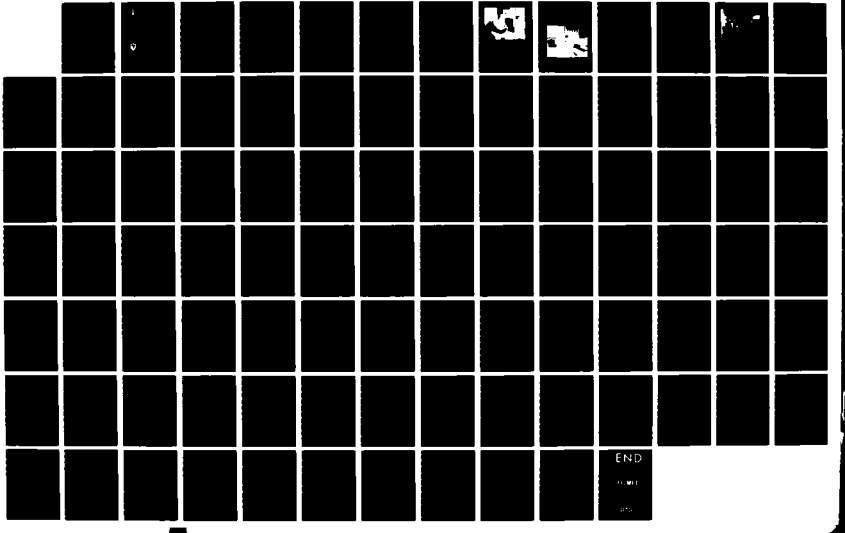
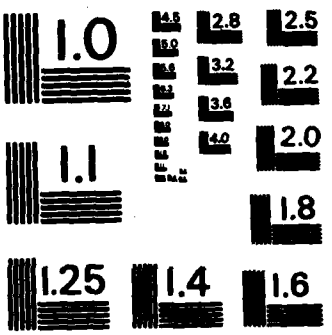
AD-A159 018 AN AUTOMATED HUMAN FACTORS ANALYSIS SYSTEM FOR IMAGING DATA(U) ARMY MISSILE COMMAND REDSTONE ARSENAL AL ADVANCED SENSORS DIR.. S R SIMS ET AL. APR 83 1/1

UNCLASSIFIED AMSMI/RE-83-24-TR SBI-AD-E950 737 F/G 17/8 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

(2

TECHNICAL REPORT RE-83-24

AN AUTOMATED HUMAN FACTORS ANALYSIS SYSTEM FOR IMAGING DATA

S. Richard Sims
Ray H. Farmer
Advanced Sensors Directorate
US Army Missile Laboratory

APRIL 1983

U.S. ARMY MISSILE COMMAND
Redstone Arsenal, Alabama 35898

Approved for public release; distribution unlimited.

DISPOSITION INSTRUCTIONS

DESTROY THIS REPORT WHEN IT IS NO LONGER NEEDED. DO NOT RETURN IT TO THE ORIGINATOR.

DISCLAIMER

THE FINDINGS IN THIS REPORT ARE NOT TO BE CONSTRUED AS AN OFFICIAL DEPARTMENT OF THE ARMY POSITION UNLESS SO DESIGNATED BY OTHER AUTHORIZED DOCUMENTS.

TRADE NAMES

USE OF TRADE NAMES OR MANUFACTURERS IN THIS REPORT DOES NOT CONSTITUTE AN OFFICIAL INDORSEMENT OR APPROVAL OF THE USE OF SUCH COMMERCIAL HARDWARE OR SOFTWARE.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER RE-83-24	2. GOVT ACCESSION NO. AD-A159018	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) An Automated Human Factors Analysis System for Imaging Data		5. TYPE OF REPORT & PERIOD COVERED Technical Report 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) S. Richard Sims and Ray H. Farmer		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Commander, US Army Missile Command ATTN: DRSMI-RE Redstone Arsenal, AL 35898		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Commander, US Army Missile Command ATTN: DRSMI-RPT Redstone Arsenal, AL 35898		12. REPORT DATE April 1983 13. NUMBER OF PAGES 114
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Automated Data Collection
Human Factors
Operator Interaction
Target Designation
FOG-M

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

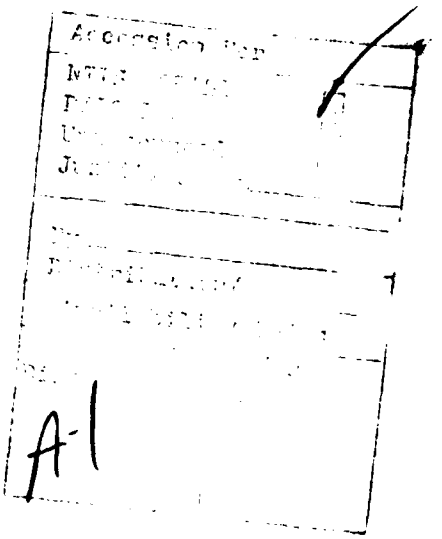
An automated human factors data collection system for imaging devices is described. This REAL-TIME collection system permits an operator to view images and react according to system requirements. Results are accumulated and stored in a VAX-11/780 computer for further processing. Target designator/cuer hardware and tracker interface circuitry designs are included. Computer program listings and sample results for a Fiber Optic Guided Missile (FOG-M) experiment are provided.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

CONTENTS



ACKNOWLEDGEMENTS

The authors hereby acknowledge the contributions of Mr. Allen E. Davis, Mr. Thomas R. Stroud, and Mr. Roy Lee on this program. Mr. Davis was responsible for the cueing hardware design and development described in Appendix E, and Mr. Stroud was responsible for the bidirectional VAX 11/780 interface design and development described in Appendix F. Mr. Davis, Mr. Stroud, and Mr. Lee assisted in obtaining the exact target locations in the video needed for the ground truth.

I. INTRODUCTION

This report describes an automated human factors data collection system for imaging devices which produce composite video. Some of the types of information available for analysis after data collection are shown in Appendix G. The system replays tapes of actual sensor video or from any other synthetic source onto a CRT screen to allow the operator to react according to individual system requirements. In general, the minimum ground truth required for each composite video data set is as follows:

1. Range IRIG time correlated with video

2. Sensor position on the range correlated with IRIG time

3. All target positions on the range correlated with IRIG time

The Fiber Optic Guided Missile (FOG-M) program instigated the design and development of this system, therefore examples will be used that relate directly to FOG-M.

II. MAN MACHINE INTERFACE

The man machine interface has always been an area of concern for systems where work load is operator intensive. In systems where an operator is required to view a CRT screen to locate targets some form of hand controller is typically used for designation. Instead of using a "joystick" or track-ball, a touch screen was used in this system to minimize the operator error and to "filter" out as many operator differences as possible. Two different types of touch screens were used in preliminary operator tests and both are available for data collection.

The first type of touch screen, the Elograph Model 270,* is a pressure sensitive type which covers the CRT screen. This touch screen allows the operator to use a ball point pen or similar stylus or his finger to designate a point on the screen.

The second type of touch screen, the Science Accessories Corporation Model GP-650,** is a sonic type which can be mounted to almost anything. With this touch screen the operator uses a pen to designate objects in the active area defined by an L frame sensor. The L frame can be sized for the application at the factory if needed.

---

1. Elographics Inc. 1976 Oak Ridge Turnpike, Oak Ridge, Tennessee 37830, (615) 482-4038

2. Science Accessories Corp. 970 King's Highway West, Southport, Connecticut 06490, (203) 255-1526

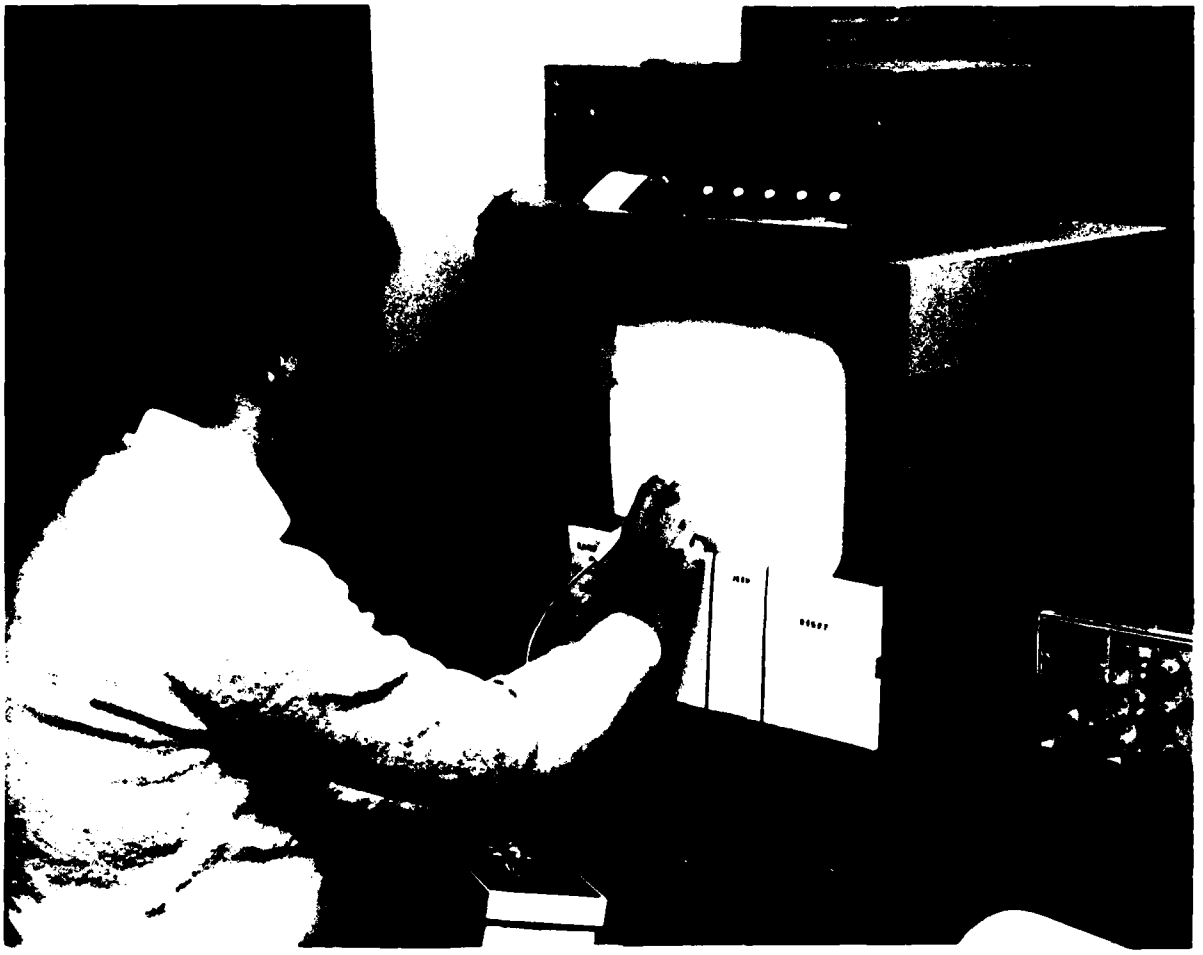


Figure 1. Touch screen used for the FOG-M human factors data collection.

<table>
<tr><td>TANK</td><td>APC</td><td rowspan="4">TRUCK</td><td rowspan="4">JEEP</td><td rowspan="4">RESET</td></tr>
<tr><td>M60</td><td>M113</td></tr>
<tr><td>M48</td><td rowspan="2">LANCE
CARRIER</td></tr>
<tr><td>M551</td></tr>
</table>

Figure 2. Target types in the FOG-M data set.

The operator station shown in Figures 3 and 4 contains:

1. An appropriate CRT display with attached touch screen.
2. Video tape player for the video source.
3. Computer terminal for controlling data collection.
4. IRIG time translator.
5. Computer interface for real time IRIG time transfer.
6. Dedicated cueing hardware to provide operator feedback and cuer testing.

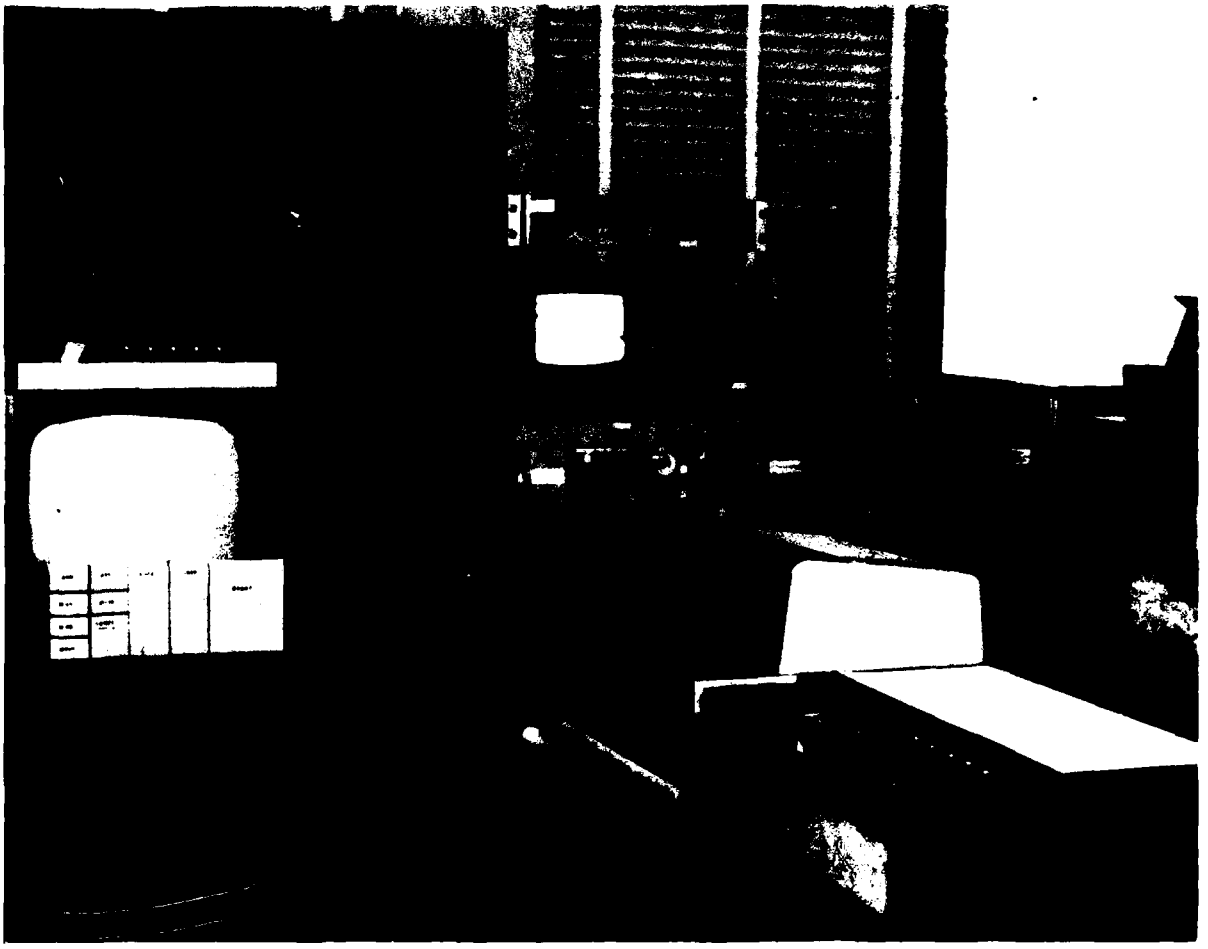


Figure 3. Operator station.

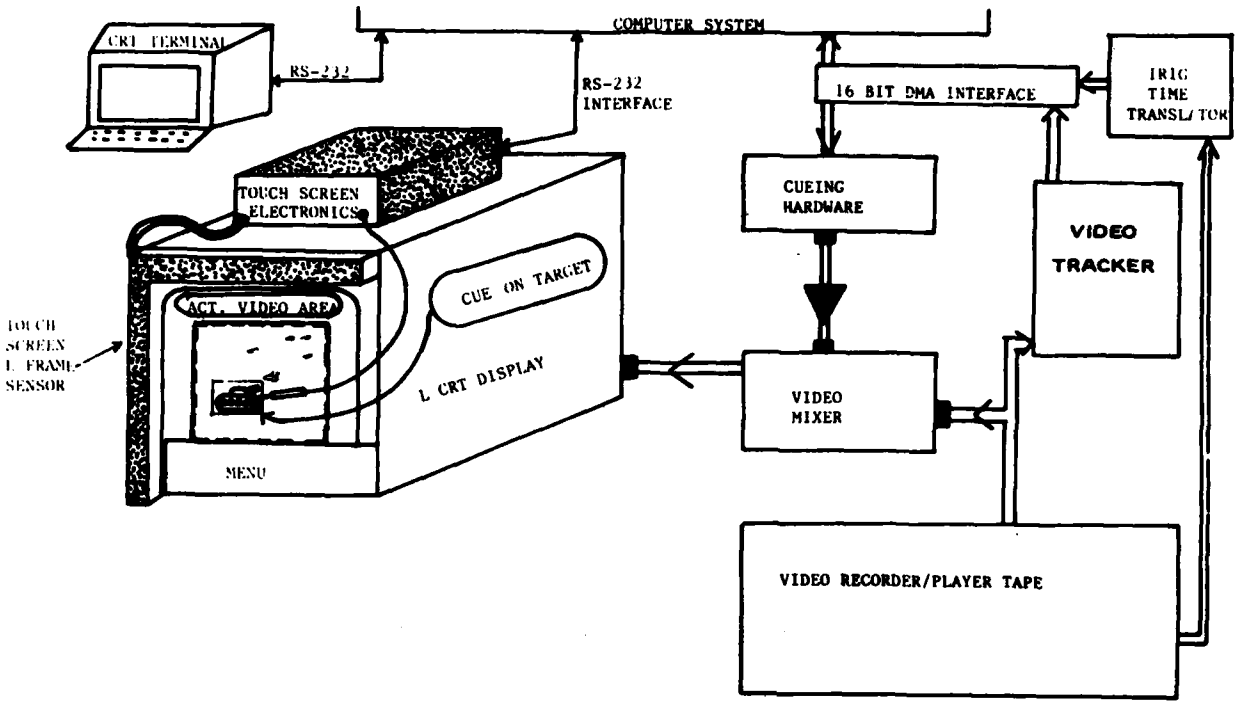


Figure 4. Operator station schematic.

III. DATA COLLECTION

Prior to beginning a data collection experiment, the operators should be given a standard briefing and allowed some finite amount of training before actually starting the experiment. Specific tasks expected of the operator should be carefully defined for the scenario under consideration. In the FOG-M application experiment, video imagery from an airborne camera flown at speeds and altitudes approximating the FOG missile trajectory was used to assess the ability of operators to find military targets such as tanks, trucks, jeeps and armored personnel carriers (APC), situated at various locations on the test ranges at Redstone Arsenal. Military and civilian personnel were selected to serve as the system operators. After a formal briefing which included descriptions of the various targets and the techniques to be used to designate the targets, each individual was placed at the operator station for familiarization with the video imagery, touch screen and the menu board. Operators were instructed to observe the display and touch the screen with the probe when (1) they discovered what might be a target, and (2) when they detected it to be an actual target. When the operators recognized the target class (tank, truck, etc.), they touched the appropriate menu area. If they were able to identify the type of tank (M60, M48, M551) or APC (M113 or Lance carrier), the menu was again used. A reset area on the menu was also available for error corrections. For each operator response the following information was stored:

1. Touch screen x coordinate

2. Touch screen y coordinate

3. IRIG time of the designation

4. Customized menu designation

Appendix A contains a listing of the data collection program used for the FOG-M operator. All responses from each operator are stored separately in individual files on a mass storage hard disk. From this information and a knowledge of the ground truth, the operators were scored on their ability to detect, recognize, and identify these targets.

IV. GROUND TRUTH GENERATION

The composite video was played back and individual targets were tracked with a video tracker in real time for ground truth generation. A program listing for the ground truth data collection program is in Appendix C. A hardware description of the tracker interface to the computer is described in Appendix F. Each target ground truth file is combined with the other target ground truths for an individual sequence on the analog tape. Some editing or additional ground truth was required on video sequences with large image movement. Each sequence of combined ground truth was then played back with ground truth cues around all targets to evaluate ground truth completeness using the program in Appendix D. The picture in Figure 5 shows an output of several cues displayed during the running of the program in Appendix D. All ground truth sequences are then sorted into one file for use by the data reduction program.

An alternate method for ground truth generation is to judiciously digitize the analog data, and on each frame, or every few frames, and where the target is virtually stationary, use those coordinates for that specific IRIG time. This later process is clearly more time consuming, but in video scenes where large accelerations and jerks are prevalent, perhaps it is the better method.

V. DATA REDUCTION AND ANALYSIS

To make the analysis as significant as possible and to minimize the statistical error, as many samples (i.e., operators) as can be generated should be used for the data collection. All individual operator results were combined into one file for input to the data reduction program. Optionally, and in addition, after each operator data run was completed, the data reduction program can be run to show the individual operator results. Appendix B contains the FOG-M data reduction program listing. The aircraft position as obtained from Mini-Ranger* data and surveyed target locations were used by the data reduction program to calculate range to target, depression angles, etc. This information was then correlated with the seeker or camera video through the common IRIG time recording. Appendix G shows the FOG-M results over the entire combined data set including all altitudes and flight directions. More specific plots and printouts of the individual altitude runs and grouped data runs have provided more detailed information, and clearly this type of

---

*Mini-Ranger is a trademark of Motorola, Inc., Government Electronics Division, Scottsdale, Arizona 85257.

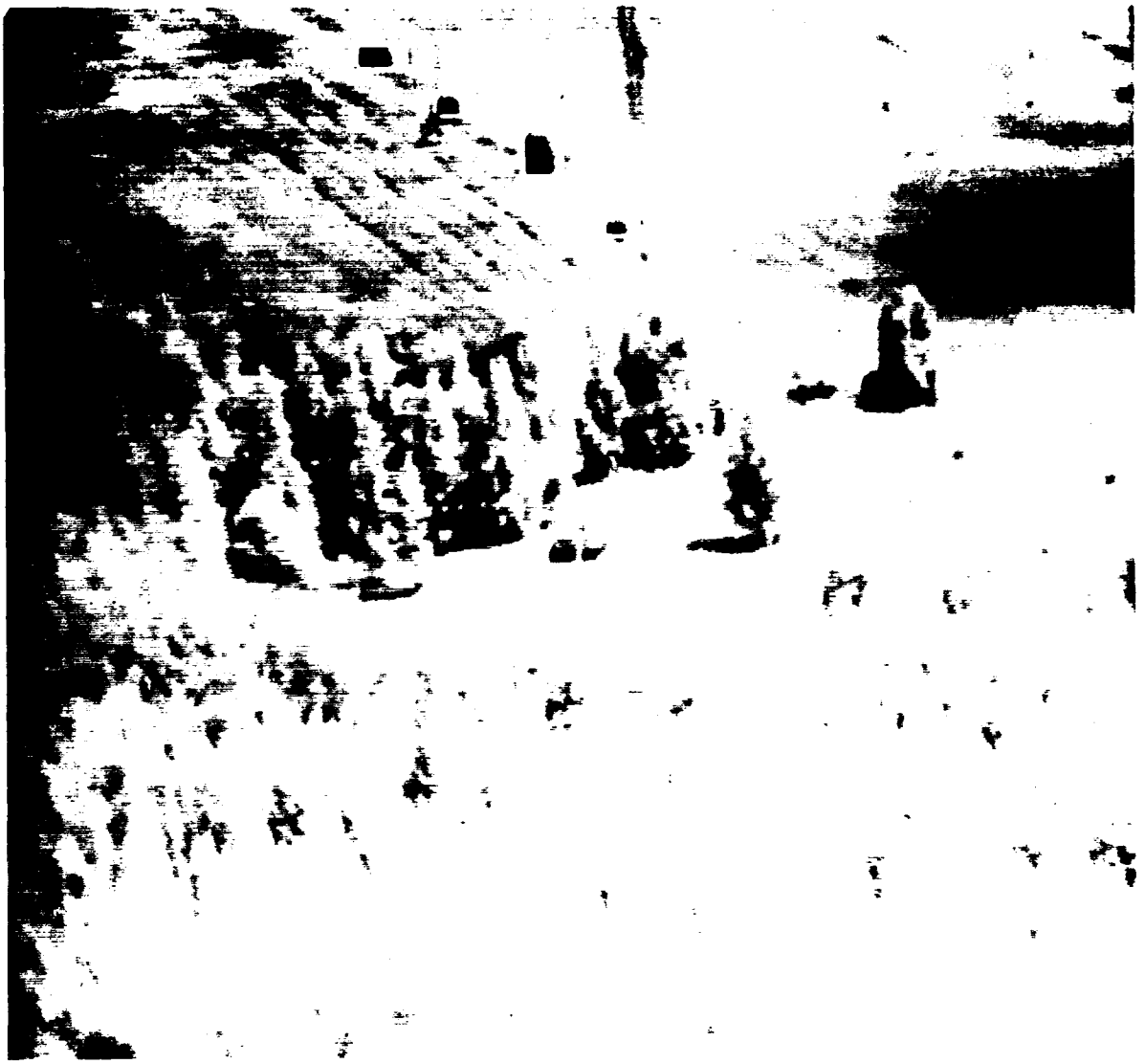


Figure 5. Picture of output of several cues displayed during the running of the program presented in Appendix D.

grouping for each specific system type will be needed to provide the detailed results needed. For complete FOG-M final data reduction and analysis results see Reference 1.

VI. CUEING HARDWARE AND EVALUATING CUEING SYSTEM PERFORMANCE

The cueing hardware is capable of generating up to ten simultaneous cues. Each cue is in the form of a box with each side independently positioned by the computer to any point on the screen. The screen resolution for the cues is 256 by 256. The intensity of the first four cues can be varied and the number one cue can be optionally "blinked."

The cueing hardware adds the additional capability of being able to evaluate automatic cuer performance. By using ground truth and adding false cues to simulate the known performance of any specific automatic cueing system operator performance can then be evaluated versus his non-cued performance. This type of evaluation will provide a means for cost/performance trade offs. Appendix E shows the cueing hardware schematics along with a definition of input/output signals. Appendix F shows and explains the computer interface hardware for use with the cueing hardware, tracker, and touch screen.

VII. PLANNED IMPROVEMENTS

A voice recognition unit will be included in the system to augment and perhaps to replace the menue on the touch screen. The intent is to allow the operator to speak the words "tank," jeep," APC," truck," or "reset" instead of having to look at and touch the menu itself. Also, a digital tracker will be included to help in ground truth generation.

VIII. CONCLUSIONS

An automated human factors analysis system for imaging data has been designed and fabricated. The equipment and computer program for the data reduction and analysis have been tested in an application requiring the detection, recognition, and identification of military targets using 26 different operators and will be used with extended data sets for the same application. How the video imagery is collected clearly will influence whether or not operator responses have any significance to the "REAL WORLD." Imagery collected to closely simulate true system parameters is obviously desirable, along with all available ground truth for the data set. The objective of this system is to show what system parameter trade-offs could be made to keep system performance at some desirable level. It is clear that analysis of operator responsivity under specific conditions can lead to reasonable system parameters which will keep overall system performance as high as possible.

APPENDIX A

FOG-M DATA COLLECTION PROGRAM USED FOR THE OPERATOR

```
      PROGRAM FOGSTATS
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     THIS PROGRAM WILL GENERATE A FOG-M HUMAN FACTORS STATISTICAL DATA
C     BASE. AN OPERATOR WILL PERFORM FOUR FUNCTIONS WHILE REAL WORLD VIDEO
C     IS DISPLAYED TO HIM TO GENERATE THIS DATA.
C
C     FUNCTION 1 -    OBJECT DESIGNATION
C     FUNCTION 2 -    TARGET DETECTION AND DESIGNATION
C     FUNCTION 3 -    TARGET RECOGNITION
C     FUNCTION 4 -    TARGET IDENTIFICATION
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     THIS PROGRAM WILL DO SOME INITIALIZATION THEN WILL BE INTERRUPTED WHEN
C     EACH TOUCH SCREEN OUTPUT IS GENERATED, OTHERWISE IT WILL HIBERNATE.
C
C**************************************************************************
C
C     IRIG(1)=FIRST IRIG TIME WORD
C     IRIG(2)=SECOND IRIG TIME WORD
C     IRIG(3)=THIRD IRIG TIME WORD
C     IRIG(4)=GATE SIZES OF THE TRACK GATE (TOP AND BOTTOM)
C     IRIG(5)=GATE SIZES OF THE TRACK GATE (LEFT AND RIGHT)
C     IRIG(6)=16 BITS THAT CONTAIN THE X AND Y POSITION OF THE CURSOR
C
C     BOXES(40) IS THE ARRAY THAT CONTAINS THE 10 BOX POSITIONS
C     LEFT,RIGHT,UP AND DOWN<-- IN THAT ORDER
C
C**************************************************************************
      EXTERNAL IO$_READVBLK,IO$_WRITEVBLK,IO$M_NOECHO
      DOUBLE PRECISION QUAD
      INTEGER SYS$WAITFR,SYS$BINTIM,SYS$HIBER,SYS$ASCTIM
      INTEGER*2 IRIG(6),IY(2),HD,TD,UD,TH,UH,TM,UM,TS1,TS2,US,LS,MS
      BYTE X,Y,XY(2),TOP,BOTTOM,LEFT,RIGHT,TB(2),LR(2)
      INTEGER*2 IOSB(4),BLINKM,ISAVE(2),TEMPP
      INTEGER SYS$QIOW,SYS$ASSIGN,SYS$QIO,SYS$LKWSET,RUN
      INTEGER INLOCK(2),IOLOCK(2),XRCHAN,ITCHAN,TSCHAN
      CHARACTER*15 TIME*23,NAME*60,RUNN*2,FNAME*13
      INTEGER*2 BOXES(40),IDAY
      CHARACTER*15 STRING
      COMMON/STRNG/STRING
      COMMON/CHAN/ITCHAN,XRCHAN,TSCHAN
      COMMON/COORD/X1,Y1,IQIOF
      COMMON/IRIGT/IRIG
      COMMON/HMSEC/IH,IM,SEC
      COMMON/BOX/BOXES
      COMMON/FLAGS/IORIGIN,IMAXY
      COMMON/DINIT/XD,YD,XORIGIN,YORIGIN
      EQUIVALENCE(XY(1),X),(XY(2),Y),(IRIG(6),XY(1))
      EQUIVALENCE(TB(1),BOTTOM),(TB(2),TOP),(IRIG(5),TB(1))
      EQUIVALENCE(LR(1),LEFT),(LR(2),RIGHT),(IRIG(4),LR(1))
      EQUIVALENCE(IRIG2,IRIG(2))
      TIME=' '
      ISTATUS=SYS$ASSIGN('XRA0',XRCHAN,,)
      IF(.NOT.ISTATUS)TYPE *,' ERROR IN XRA1 CHANNEL ASSIGN'
```

```
                ISTATUS=SYS$ASSIGN('TTB6',TSCHAN,,)
                IF(.NOT.ISTATUS)TYPE *,' ERROR IN TTB6 CHANNEL ASSIGN'
                IFLAG=1
                IMLOCK(1)=%LOC(BOXES(1))
                IMLOCK(2)=%LOC(BOXES(40))
                K=SYS$LKWSET(IMLOCK,IOLOCK,)
                IF(.NOT.K)TYPE *,' UNABLE TO LOCK BOXES I/O BUFFER'
                IMLOCK(1)=%LOC(IRIG(1))
                IMLOCK(2)=%LOC(IRIG(6))
                K=SYS$LKWSET(IMLOCK,IOLOCK,)
                IF(.NOT.K)TYPE *,' UNABLE TO LOCK IRIG I/O BUFFER'
                IMLOCK(1)=%LOC(STRING(1:1))
                IMLOCK(2)=%LOC(STRING(16:16))
                K=SYS$LKWSET(IMLOCK,IOLOCK,)
                IF(.NOT.K)TYPE *,' UNABLE TO LOCK IRIG I/O BUFFER'
                OPEN(UNIT=6,NAME='TT',STATUS='OLD')
                TYPE *,' ENTER VIDEO TAPE NUMBER BEING VIEWED(TAPE 1 OR 2)'
                READ(6,66)IVIDEOT
       66       FORMAT(I1)
                OPEN(UNIT=12,NAME='OPERATOR.FOR',TYPE='NEW')
                WRITE(12,66)IVIDEOT
       66       FORMAT(1X,'TAPE NUMBER ',I1,' FOR-M OPERATOR DATA FOLLOWS')
       C        K=SYS$ASCTIM(,TIME,,,)
                TYPE *,' TEST THE TOUCH SCREEN FOR I/O ENABLED.................'
                K = SYS$QIOW(%VAL(1),%VAL(TSCHAN),
               2%VAL(%LOC(IO$_READVBLK).OR.%LOC(IO$M_NOECHO)),
               1IOSB,,,DATA,%VAL(15),,,,)
                TYPE *,' TOUCH SCREEN NOW READY FOR OPERATOR INPUT.',
               1' IS IRIG UNIT IN TRANSLATE??'
       C        XORIGIN=5.6
                XORIGIN=5.62
                YORIGIN=4.87
                XS=20.48
                YS=24.0
                XD=XS-XORIGIN
                YD=YS-YORIGIN
                CALL UIO
       1        CONTINUE
                K=SYS$HIBER()
                GO TO 1
                 END
                SUBROUTINE UIO
                BYTE C
                EXTERNAL IO$_READVBLK,IO$M_NOECHO
                EXTERNAL INPUT
                INTEGER SYS$ASSIGN,SYS$QIOW,ITCHAN,SYS$QIO,XRCHAN
                INTEGER IO$_READVBLK,IO$M_NOECHO,TSCHAN
                CHARACTER*15 STRING
                BYTE DATA(15)
                INTEGER*2 IOSB(4)
                EQUIVALENCE(DATA,STRING)
                COMMON/CHAN/ITCHAN,XRCHAN,TSCHAN
                COMMON/DINIT/XD,YD,XORIGIN,YORIGIN
                COMMON/STRNG/STRING
                COMMON/FLAGS/IORIGIN,IMAXY
```

```
      DATA IORIGIN/1/,IMAXY/1/
      K = SYS$QIO(%VAL(5),%VAL(TSCHAN),%VAL(%LOC(IO$_READVBLK)),IOSB,
      1INPUT,C,
      2 DATA(1),%VAL(15),,,,)
      RETURN
      END
      SUBROUTINE INPUT(C)
      BYTE C,XY(2)
      EXTERNAL IO$_READVBLK,IO$M_NOECHO,IO$_WRITEVBLK
      INTEGER SYS$QIOW,ITCHAN,XRCHAN,IO$M_NOECHO,SYS$SETIMR
      INTEGER SYS$QIO,SYS$WAITFR,TSCHAN,SYS$BINTIM
      DOUBLE PRECISION QUAD
      INTEGER*2 IRIG(6),IY(2),HD,TD,UD,TH,UH,TM,UM,TS1,TS2,US,LS,MS
      INTEGER*2 ISAVE(2)
      CHARACTER*15 STRING,TIME*16
      INTEGER*2 BOXES(40)
      INTEGER*2  IOSB(4)
      BYTE DATA(15)
C     THESE PARAMETERS ARE LARGER THAN THE SCALE FACTORS SO YOU CAN
C     GO OUTSIDE THE RASTER AREA ALSO.
      COMMON/IRIGT/IRIG
      COMMON/CHAN/ITCHAN,XRCHAN,TSCHAN
      COMMON/DINIT/XD,YD,XORIGIN,YORIGIN
      COMMON/COORD/X,Y,IQIOF
      COMMON/BOX/BOXES
      COMMON/STRNG/STRING
      COMMON/HMSEC/IH,IM,SEC
      COMMON/FLAGS/IORIGIN,IMAXY
      EQUIVALENCE(DATA,STRING)
      EQUIVALENCE(IRIG(6),XY(1))
C     TYPE *,' ENTERING AST INTERRUPT ROUTINE'
      IQIOF=1
      IBADV=0
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C     THIS PROGRAM READS THE TOUCH SCREEN DIGITIZER PUTS THE CRT
C     COORDINATES IN COMMON /COORD/
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
776   K=SYS$QIOW(%VAL(4),%VAL(XRCHAN),%VAL(%LOC(IO$_READVBLK)),IOSB,,,
      1 IRIG(1),%VAL(12),,,,)
      IF(.NOT.K)TYPE *,' ERROR ON SYS$QIO READING IRIG TIME'
      IF(IRIG(3))THEN
            IF(IBADV.EQ.1)GO TO 776
      IBADV=1
      WRITE(6,157)X,Y,IDAY,IH,IM,SEC
157   FORMAT(1X,F8.2,1X,F8.2,5X,I3,':',I2,':',I2,':',F6.2,
      1' INVALID IRIG FROM VIDEO TAPE')
      GO TO 776
      ENDIF
      IF(IBADV.EQ.1)CALL UIO
      HD=ISHFT(IRIG(1),-12)
      TD=IAND(ISHFT(IRIG(1),-8),'F'X)
      UD=IAND(ISHFT(IRIG(1),-4),'F'X)
```

```
166                 TH=IAND(IRIG(1),'3'X)
167                 UH=ISHFT(IRIG(2),-12)
168                 TM=IAND(ISHFT(IRIG(2),-9),'7'O)
169                 UM=IAND(ISHFT(IRIG(2),-5),'F'X)
170                 TS1=IAND(ISHFT(IRIG(2),-2),'7'O)
171                 ISAVE(1)=IAND(ISHFT(IRIG(2),2),'C'X)
172                 ISAVE(2)=ISHFT(IRIG(3),-14)
173                 US=IOR(ISAVE(1),ISAVE(2))
174                 TS2=IAND(ISHFT(IRIG(3),-10),'F'X)
175                 LS=IAND(ISHFT(IRIG(2),-6),'F'X)
176                 MS=IAND(ISHFT(IRIG(3),-2),'F'X)
177                 IDAY=100*HD+10*TD+UD
178                 IH=TH*10+UH
179                 IM=TM*10+UM
180                 SEC=FLOAT(TS1*10+US)+FLOAT(TS2)/10.+FLOAT(LS)/100.
181                 1+FLOAT(MS)/1000.
182           700   K=SYS$WAITFR(%VAL(5))
183                 IF(.NOT.K)TYPE*,' ERROR ON WAITFR IN AST ROUTINE, %S'
184                 DECODE(14,701,STRING,ERR=772)X,Y
185           701   FORMAT(1X,F6.2,2X,F6.2)
186           1     CONTINUE
187           C     WRITE(6,19)X,Y,XD,XORIGIN,YORIGIN
188           19    FORMAT(' X=',F8.2,' Y=',F6.2,' XD=',F6.2,' XORIGIN=',F6.2,
189                 1' YORIGIN=',F6.2)
190                 IF(X-XORIGIN.EQ.0.0)THEN
191                 X=0.0
192                 GO TO 23
193                 ENDIF
194                 IF(Y-YORIGIN.EQ.0.0)THEN
195                 Y=0.0
196                 GO TO 25
197                 ENDIF
198                 X=254./((XD/(X-XORIGIN)))
199                 Y=242./((YD/(Y-YORIGIN)))
200                 GO TO 25
201           23    IF(Y-YORIGIN.NE.0.0)THEN
202                 Y=242./((YD/(Y-YORIGIN)))
203                 ELSE
204                 Y=0.0
205                 ENDIF
206           C     WRITE(6,18)X,Y,XD,XORIGIN,YORIGIN
207           18    FORMAT(' X=',F10.2,' Y=',F10.2,' XD=',F6.2,' XORIGIN=',F6.2,
208                 1' YORIGIN=',F6.2)
209           C     245.-252.-8.  WHICH ARE THE MAX X AND MIN X VISIBLE OF THE BOX
210           C     241.-253.-12. WHICH ARE THE MAX Y AND MIN Y VISIBLE OF THE BOX
211           25    IF(Y.GT.254.)THEN
212                                 IF(X.GE.95..AND.X.LT.167.)THEN
213                                 WRITE(12,61)X,Y,IDAY,IH,IM,SEC
214                                 WRITE(6,61)X,Y,IDAY,IH,IM,SEC
215           61    FORMAT(1X,F8.2,1X,F8.2,5X,I3,':',I2,':',I2,':',F6.2,
216                 1' TRUCK RECOGNITION')
217                                 GO TO 26
218                                 ENDIF
219                                 IF(X.GE.167..AND.X.LT.237.)THEN
220                                 WRITE(12,62)X,Y,IDAY,IH,IM,SEC
```

```
                        WRITE(6,62)X,Y,IDAY,IN,IM,SEC
   62   FORMAT(1X,F8.2,1X,F8.2,5X,I3,':',I2,':',I2,':',F6.2,
     1' JEEP RECOGNITION')
                        GO TO 26
                        ENDIF
                        IF(X.GE.237.)THEN
                        X=255.
                        Y=255.
                        WRITE(12,63)X,Y,IDAY,IN,IM,SEC
                        WRITE(6,63)X,Y,IDAY,IN,IM,SEC
   63   FORMAT(1X,F8.2,1X,F8.2,5X,I3,':',I2,':',I2,':',F6.2,
     1' ******** RESET ******')
                        GO TO 26
                        ENDIF
                IF(Y.LT.306.)THEN
                        IF(X.LT.27.)THEN
                        WRITE(12,58)X,Y,IDAY,IN,IM,SEC
                        WRITE(6,58)X,Y,IDAY,IN,IM,SEC
   58   FORMAT(1X,F8.2,1X,F8.2,5X,I3,':',I2,':',I2,':',F6.2,
     1' TANK RECOGNITION')
                        GO TO 26
                        ENDIF
                        IF(X.GE.27..AND.X.LT.94.7)THEN
                        WRITE(12,60)X,Y,IDAY,IN,IM,SEC
                        WRITE(6,60)X,Y,IDAY,IN,IM,SEC
   60   FORMAT(1X,F8.2,1X,F8.2,5X,I3,':',I2,':',I2,':',F6.2,
     1' APC RECOGNITION')
                        GO TO 26
                        ENDIF
                        ENDIF
        IF(Y.GE.306..AND.Y.LT.354)THEN
                        IF(X.LT.27.)THEN
                        WRITE(12,64)X,Y,IDAY,IN,IM,SEC
                        WRITE(6,64)X,Y,IDAY,IN,IM,SEC
   64   FORMAT(1X,F8.2,1X,F8.2,5X,I3,':',I2,':',I2,':',F6.2,
     1' M60 IDENTIFICATION')
                        GO TO 26
                        ENDIF
                        IF(X.GE.27..AND.X.LT.94.7)THEN
                        WRITE(12,65)X,Y,IDAY,IN,IM,SEC
                        WRITE(6,65)X,Y,IDAY,IN,IM,SEC
   65   FORMAT(1X,F8.2,1X,F8.2,5X,I3,':',I2,':',I2,':',F6.2,
     1' M113 IDENTIFICATION')
                        GO TO 26
                        ENDIF
                        ENDIF
        IF(Y.GE.354..AND.Y.LT.396.)THEN
                        IF(X.LT.27.)THEN
                        WRITE(12,66)X,Y,IDAY,IN,IM,SEC
                        WRITE(6,66)X,Y,IDAY,IN,IM,SEC
   66   FORMAT(1X,F8.2,1X,F8.2,5X,I3,':',I2,':',I2,':',F6.2,
     1' M48 IDENTIFICATION')
                        GO TO 26
                        ENDIF
                        IF(X.GE.27..AND.X.LT.94.7)THEN
```

```
                                WRITE(12,67)X,Y,IDAY,IN,IN,SEC
                                WRITE(6,67)X,Y,IDAY,IN,IN,SEC
67      FORMAT(1X,F8.2,1X,F8.2,6X,I3,':',I2,':',I2,':',F6.2,
     1' LANCE CARRIER IDENTIFICATION')
                                GO TO 26
                                ENDIF
                                ENDIF
        IF(Y.GE.396.)THEN
                                IF(X.LT.27.)THEN
                                WRITE(12,68)X,Y,IDAY,IN,IN,SEC
                                WRITE(6,68)X,Y,IDAY,IN,IN,SEC
68      FORMAT(1X,F8.2,1X,F8.2,6X,I3,':',I2,':',I2,':',F6.2,
     1' M551 IDENTIFICATION')
                                GO TO 26
                                ENDIF
                                IF(X.GE.27..AND.X.LT.94.7)THEN
                                WRITE(12,67)X,Y,IDAY,IN,IN,SEC
                                WRITE(6,67)X,Y,IDAY,IN,IN,SEC
C       LANCE CARRIER ID AGAIN
                                GO TO 26
                                ENDIF
                                ENDIF
        ENDIF
        WRITE(12,56)X,Y,IDAY,IN,IN,SEC
        WRITE(6,56)X,Y,IDAY,IN,IN,SEC
56      FORMAT(1X,F8.2,1X,F8.2,6X,I3,':',I2,':',I2,':',F6.2,
     1' OBJECT OR TARGET DESIGNATION')
26      BOXES(1)=0
        BOXES(2)=0
        BOXES(3)=0
        BOXES(4)=0
        BOXES(5)=X-1.
        IF(BOXES(5).LT.0)BOXES(5)=IAND(BOXES(5),'377'O)
        IF(BOXES(5).LT.1)BOXES(5)=1
        IF(BOXES(5).GT.254)BOXES(5)=254
        BOXES(6)=X+1.
        IF(BOXES(6).LT.0)BOXES(6)=IAND(BOXES(6),'377'O)
        IF(BOXES(6).GT.254)BOXES(6)=254
        BOXES(7)=Y-1.
        IF(BOXES(7).LT.0)BOXES(7)=IAND(BOXES(7),'377'O)
        IF(BOXES(7).LT.1)BOXES(7)=1
        IF(BOXES(7).GT.242)BOXES(7)=242
        BOXES(8)=Y+1.
        IF(BOXES(8).LT.0)BOXES(8)=IAND(BOXES(8),'377'O)
        IF(BOXES(8).GT.242)BOXES(8)=242
C       WRITE(6,101)BOXES(5),BOXES(6),BOXES(7),BOXES(8)
101     FORMAT(1X,4(2X,O6))
        K=SYSSWAITFR(%VAL(4))
        IF(.NOT.K)TYPE*,' ERROR ON WAITFR IN AST ROUTINE, #4'
        K = SYSSQIO(%VAL(4),%VAL(%RCHAN),%VAL(%LOC(IO$_WRITEVBLK)),
     1IOSB,,,BOXES(1),%VAL(8),,,,)
100     CONTINUE
        CALL UIO
        RETURN
772     TYPE *,' ERROR IN DECODE OF THE DIGITIZER OUTPUT'
```

```
331                     K = SYSQIO(%VAL(5),%VAL(TSCHAN),%VAL(%LOC(IOS_READVBLK)),IOSB,
332                     1,,
333                     2 DATA(1),%VAL(16),,,,)
334                     GO TO 700
335                     END
```

APPENDIX B

FOG-M DATA REDUCTION PROGRAM LISTING

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C        THIS PROGRAM READS THE [.ATVS]OPERATOR.TOT FILE WHICH IS THE
C        CONCATENATED OPERATOR.FOG FILES.
C
C        THE MINI.DAT FILE CONTAINS ALL THE ATVS MINIRANGER DATA FOR
C        SEQUENCE 1 THROUGH SEQUENCE 18.
C
C        THE GROUND.FFF FILE CONTAINS ALL THE GROUND TRUTH TO DATE ON THE
C        EIGHTEEN SEQUENCES. THIS FILE IS SORTED BY TOTAL IRIG.
C
C
C        BEFORE RUNNING THIS PROGRAM THE OPERATOR.TOT FILE NEEDS TO BE
C        GENERATED USING THE FOLLOWING COMMAND IF IS DOESN'T ALREADY EXIST:
C
C                $APPEND OPERATOR.FOG;* OPERATOR.TOT
C
C        THIS PROGRAM WILL TAKE THE OPERATOR.TOT DATA FILE AND GENERATE
C        MEAN DELTA T'S AND STANDARD DEVIATIONS BETWEEN:
C
C                        1. - OBJECT DESIGNATION AND TARGET DETECTION
C                        2. - TARGET DETECTION   AND TARGET RECOGNITION
C                        3. - TARGET RECOGNITION AND TARGET IDENTIFICATION
C*********************************************************************
C                        4. - OBJECT DESIGNATION RANGES HISTOGRAM
C                        5. - TARGET DETECTION RANGES HISTOGRAM
C                        6. - TARGET RECOGNITION RANGES HISTOGRAM
C                        7. - TARGET IDENTIFICATION RANGES HISTOGRAM
C
C                NOTE:   4 THROUGH 7 CAN ONLY BE DONE ONE AT A TIME BY
C                        SELECTING WITH THE SPECIFIC CALL TO HIST2 BELOW.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        PROGRAM REDUCE4
        LOGICAL*1 ODFLAG,ORFLAG,OIFLAG
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        REAL HDELTAD(4000),HDELTAR(4000),HDELTAI(4000),HDELTAO(4000)
        BYTE ITANKID(4000),ITANKIR(4000),ITANKII(4000)
C
C       HDELTAO IS THE DATA ARRAY FOR THE HISTOGRAM OBJECT DESIGNATION RANGES
C       HDELTAD IS THE DATA ARRAY FOR THE HISTOGRAM TARGET DETECTION   RANGES
C       HDELTAR IS THE DATA ARRAY FOR THE HISTOGRAM TARGET RECOGNITION RANGES
C       HDELTAI IS THE DATA ARRAY FOR THE HISTOGRAM TARGET IDENTIFICATION RANGES
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        PARAMETER ERRORMINI=1.5 !ALLOWABLE MINI RANGER TIME DELTA FOR RANGE CALCULATIONS
        PARAMETER MEM1=80000
        CHARACTER*40 TEXT,NAME*20,TAPE*45,CODE*2,NAME1*15
        INTEGER*2 IDAY,TIDAY(MEM1),SEQS(19),ALTT(3)
        INTEGER ALTS(4),TANKR,APCR,TRUCKR,JEEPR,M60ID,M48ID,M551ID,M113ID
        INTEGER LANCEID,RESETS,FRECOG,FALSEID,FRECOGF,FALSEIDF
        INTEGER WTANKR,WAPCR,WTRUCKR,WJEEPR,WM60ID,WM48ID,WM551ID,WM113ID
        INTEGER WLANCEID,ITARGETN
        BYTE TIH(MEM1),TIM(MEM1),IH,IM,VALID
        REAL TSEC(MEM1),TX(MEM1),TY(MEM1),TZ(MEM1)
        COMMON/COUNTS/IFSEQ4,IFSEQ13,IFSEQ17,SEQS
```

```
      COMMON/TANKS/XTANK(13),YTANK(13),ITOTALT
C     ALTS(1) ARE THE 325 AND 330 RUNS
C     ALTS(2) ARE THE 500          RUNS
C     ALTS(3) ARE THE 825 AND 830 RUNS
C     ALTS(4) ARE THE RUNS NOT COVERED BY 1 ,2, OR 3
      DATA ALTS/4*0/,ALTT/325,500,800/
      DATA XTANK/1795..1748.,1713.,1773.,1820.,1974.5,2026.5,2073.2,
      1212*0.,973.,973.,973.,973./
      DATA YTANK/-20.,-110.,-159.,-170.,-140.,-123.,-123.,-123.,-123.,
      1-327.,-276.,-235.,-196./
      CALL GTRUTH(IDAY,IH,IM,SEC,X,Y,VALID,ITARGETN)
C     TYPE *,' ENTER FILE NAME.(i.e. OPERATOR.TOT)'
C     READ(5,56)NAME
56    FORMAT(A)

      NAME='[.ATVS]OPERATOR.TOT'

      OPEN(UNIT=12,NAME=NAME,TYPE='OLD',READONLY,SHARED)
1212  FORMAT('0')
      NAME1='MINI.DAT'
      TYPE *,' READING UNFORMATTED MINI RANGER DATA FROM FILE ',NAME1
      OPEN(UNIT=9,NAME=NAME1,TYPE='OLD',FORM='UNFORMATTED',
      1SHARED,READONLY)
      IL=0
500   READ (9,END=4555,ERR=545) IDAY,IH,IM,SEC,X,Y,Z
C     WRITE(6,100)IH,IM,SEC,X,Y,Z
100   FORMAT(1X,I2,':',I2,':',F6.3,3X,3(F9.1,1X))
      IL=IL+1
      IF(IL.GT.MEM1)
      1TYPE *,' ERROR IN ARRAY DIMENSIONS. MAKE MEM1 BIGGER IL=',IL
      TIDAY(IL)=IDAY
      TIH(IL)=IH
      TIM(IL)=IM
      TSEC(IL)=SEC
      TX(IL)=X
      TY(IL)=Y
      TZ(IL)=Z
      GO TO 500
545   TYPE *,' ERROR ON KINGAIR MINIRANGER DATA READ ',IDAY,IH
      STOP
4555  TYPE *,IL,' MINI RANGER DATA READ SO LETS GET ON WITH IT.'
C     READ(12,67)TAPE
67    FORMAT(A)
C     TYPE *,TAPE
      WTANKR=0
      WAPCR=0
      WTRUCKR=0
      WJEEPR=0
      WM60ID=0
      WM48ID=0
      WM113ID=0
      WM551ID=0
      WLANCEID=0
      NOPER=0
      IHDELTAO=0
```

```
111                     INDELTAD=0
112                     INDELTAR=0
113                     INDELTAI=0
114                     IDCOUNT=0
115                     DELTAD=0.0
116                     DELTADS=0.0
117                     IRCOUNT=0
118                     DELTAR=0.0
119                     DELTARS=0.0
120                     IICOUNT=0
121                     DELTAI=0.0
122                     DELTAIS=0.0
123                     FRECOGF=0
124                     FALSEIDF=0
125                     FRECOG=0
126                     FALSEID=0
127                     TANKR=0
128                     APCR=0
129                     TRUCKR=0
130                     JEEPR=0
131                     M60ID=0
132                     M48ID=0
133                     M113ID=0
134                     LANCEID=0
135                     M88IID=0
136                     RESETS=0
137                     IOPIN=0
138                     DTIME=777777.
139                     RTIME=777777.
140             1       READ(12,60,ERR=7676,END=777)X,Y,IDAY,IHH,IMM,SEC,TEXT
141                     IOPIN=IOPIN+1
142             60      FORMAT(1X,F8.2,1X,F8.2,6X,I3,1X,I2,1X,I2,1X,F6.2,A)
143             C       WRITE(6,60)X,Y,IDAY,IHH,IMM,SEC,TEXT
144                     CALL GTRUTH(IDAY,IHH,IMM,SEC,X,Y,VALID,ITANKI)
145                     CODE=TEXT(2:3)
146                     IDSEARCH=INDEX(TEXT,'IDENT')
147                     IF(.NOT.VALID)THEN
148             C       WRITE(6,60)X,Y,IDAY,IHH,IMM,SEC,TEXT
149                     GO TO 1
150                     ENDIF
151             CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
152             C
153             C       RESET ERROR IN OPERATOR.FOR FIX. OPERATOR.FOR HAS ALSO BEEN FIXED
154             C
155                     IF(CODE.EQ.'JE'.AND.X.EQ.255..AND.Y.EQ.255.)CODE='**'
156             CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
157                     IF(CODE.EQ.'**')THEN
158                     RESETS=RESETS+1
159             C       ITANKI=0
160                     GO TO 1
161                     ENDIF
162                     IF(ITANKI.EQ.0.AND.VALID.EQ.3)THEN
163                             IF(IDSEARCH.EQ.0)THEN
164                                     FRECOGF=FRECOGF+1
165                             ELSE
```

```
166                                 FALSEIDF=FALSEIDF+1
167                         ENDIF
168                 GO TO 1
169                 ENDIF
170                 TIME =FLOAT(IDAY-215)*60.*60.*60.+
171                 1FLOAT(IHH )*60.*60.+FLOAT(IMM )*60.+SEC
172           C     WRITE(6,618)X,Y,IDAY,IHH,IMM,SEC,TEXT,ITANKI
173           618   FORMAT(1X,F8.2,1X,F8.2,5X,I3,1X,I2,1X,I2,1X,F6.2,A,5X,I2)
174                 IF(CODE.EQ.'OB')THEN
175           C             TYPE *,' VALID TARGET DETECTION'          !********
176                         IHDELTAD=IHDELTAD+1
177                         IHDELTAO=IHDELTAO+1
178           C                 TYPE*,'IHDELTA=',IHDELTA,'ITANKI=',ITANKI,'TIME=',TIME
179                         ITANKID(IHDELTAD)=ITANKI
180                         HDELTAD(IHDELTAD)=TIME
181                                 IF(ABS(TIME-DTIME).LT.10.)THEN
182                                 DELTAD=DELTAD+(TIME-DTIME)
183                                 DELTADS=DELTADS+(TIME-DTIME)**2
184                                 IDCOUNT=IDCOUNT+1
185                                 ENDIF
186                         DTIME=TIME      ! TARGET DETECT TIME
187                         IDDAY=IDAY
188                         GO TO 1
189                 ENDIF
190                 IF(CODE.EQ.'TA'.OR.CODE.EQ.'AP'.OR.CODE.EQ.'TR'.OR.CODE.EQ.'JE')THEN
191                 IF(IDSEARCH.NE.0)GO TO 10
192                         RTIME=TIME
193           C             TYPE *,' VALID TARGET RECOGNITION'        !*******
194                 ENDIF
195                 IF(CODE.EQ.'TA')THEN
196                         IF(ITANKI.EQ.2.OR.
197                 1         ITANKI.EQ.3.OR.
198                 1         ITANKI.EQ.4.OR.
199                 1         ITANKI.EQ.5.OR.
200                 1         ITANKI.EQ.10.OR.
201                 1         ITANKI.EQ.11.OR.
202                 1         ITANKI.EQ.12.OR.
203                 1         ITANKI.EQ.13)THEN
204                         IHDELTAR=IHDELTAR+1
205                         HDELTAR(IHDELTAR)=TIME
206                         TANKR=TANKR+1
207                         ELSE
208                         WTANKR=WTANKR+1
209                         GO TO 11
210                         ENDIF
211                 GO TO 2
212                 ENDIF
213                 IF(CODE.EQ.'AP')THEN
214                         IF(ITANKI.EQ.1.OR.
215                 1         ITANKI.EQ.8.OR.
216                 1         ITANKI.EQ.9)THEN
217                         IHDELTAR=IHDELTAR+1
218                         HDELTAR(IHDELTAR)=TIME
219                         APCR=APCR+1
220                         ELSE
```

```
                        WAPCR=WAPCR+1
                        GO TO 11
                        ENDIF
                GO TO 2
                ENDIF
                IF(CODE.EQ.'TR')THEN
                        IF(ITANKI.EQ.7)THEN
                        IHDELTAR=IHDELTAR+1
                        HDELTAR(IHDELTAR)=TIME
                        TRUCKR=TRUCKR+1
                        ELSE
                        WTRUCKR=WTRUCKR+1
                        GO TO 11
                        ENDIF
                GO TO 2
                ENDIF
                IF(CODE.EQ.'JE')THEN
                        IF(ITANKI.EQ.6)THEN
                        IHDELTAR=IHDELTAR+1
                        HDELTAR(IHDELTAR)=TIME
                        JEEPR=JEEPR+1
                        ELSE
                        WJEEPR=WJEEPR+1
                        GO TO 11
                        ENDIF
                GO TO 2
                ENDIF
                GO TO 10
        2       IF(ABS(TIME-DTIME).LT.10.)THEN
                DELTAR=DELTAR+(TIME-DTIME)
                ITANKIR(IHDELTAR)=ITANKI
                IRCOUNT=IRCOUNT+1
                DELTARS=DELTARS+(TIME-DTIME)**2
                ENDIF
                GO TO 1
        11      FRECOG=FRECOG+1
        C       FALSE RECOGNITION OF AN ACTUAL TARGET
                GO TO 1
        10      IF(CODE.EQ.'M6'.OR.CODE.EQ.'M4'.OR.CODE.EQ.'M5'.OR.CODE.EQ.
                1'M1'.OR.CODE.EQ.'LA')THEN
                ENDIF
                IF(CODE.EQ.'M6')THEN
                        IF(ITANKI.EQ.2)THEN
                        M60ID=M60ID+1
                        IHDELTAI=IHDELTAI+1
                        HDELTAI(IHDELTAI)=TIME
                        ELSE
                        WM60ID=WM60ID+1
                        GO TO 12
                        ENDIF
                GO TO 3
                ENDIF
                IF(CODE.EQ.'M4')THEN
                        IF(ITANKI.EQ.5.OR.
                1          ITANKI.EQ.10.OR.
```

```
276                     1           ITANKI.EQ.11.OR.
277                     1           ITANKI.EQ.12.OR.
278                     1           ITANKI.EQ.13)THEN
279                             IHDELTAI=IHDELTAI+1
280                             HDELTAI(IHDELTAI)=TIME
281                             M48ID=M48ID+1
282                             ELSE
283                             WM48ID=WM48ID+1
284                             GO TO 12
285                             ENDIF
286                     GO TO 3
287                     ENDIF
288                     IF(CODE.EQ.'M5')THEN
289                             IF(ITANKI.EQ.3.OR.
290                     1          ITANKI.EQ.4)THEN
291                             IHDELTAI=IHDELTAI+1
292                             HDELTAI(IHDELTAI)=TIME
293                             M551ID=M551ID+1
294                             ELSE
295                             WM551ID=WM551ID+1
296                             GO TO 12
297                             ENDIF
298                     GO TO 3
299                     ENDIF
300                     IF(CODE.EQ.'M1')THEN
301                             IF(ITANKI.EQ.1.OR.
302                     1          ITANKI.EQ.8)THEN
303                             IHDELTAI=IHDELTAI+1
304                             HDELTAI(IHDELTAI)=TIME
305                             M113ID=M113ID+1
306                             ELSE
307                             WM113ID=WM113ID+1
308                             GO TO 12
309                             ENDIF
310                     GO TO 3
311                     ENDIF
312                     IF(CODE.EQ.'LA')THEN
313             C       TYPE *,'CODE=',CODE,' ITANK=',ITANKI
314                             IF(ITANKI.EQ.9)THEN
315                             IHDELTAI=IHDELTAI+1
316                             HDELTAI(IHDELTAI)=TIME
317                             LANCEID=LANCEID+1
318                             ELSE
319                             VLANCEID=VLANCEID+1
320                             GO TO 12
321                             ENDIF
322                     GO TO 3
323                     ENDIF
324                     GO TO 15
325             3       IF(ABS(TIME-RTIME).LT.10.)THEN
326                     DELTAI=DELTAI+(TIME-RTIME)
327                     ITANKII(IHDELTAI)=ITANKI
328                     IICOUNT=IICOUNT+1
329                     DELTAIS=DELTAIS+(TIME-RTIME)**2
330                     ENDIF
```

```
331                 GO TO 1
332           12    FALSEID=FALSEID+1
333           C     WRONG IDENTIFICATION OF ACTUAL TARGET
334                 GO TO 1
335           15    TYPE *,CODE
336                 STOP ' ERROR ON CODE'
337           777   CONTINUE
338                 IOPIN=IOPIN-NOPER
339                 WRITE(6,37676)TANKR,WTANKR,APCR,WAPCR,TRUCKR,WTRUCKR,JEEPR,WJEEPR,
340                 1M60ID,WM60ID,M48ID,WM48ID,M551ID,WM551ID,M113ID,WM113ID,
341                 1LANCEID,WLANCEID,
342                 2RESETS,NOPER,IOPIN
343           37676 FORMAT(1X,'NUMBER OF CORRECT TANK  RECOGNITIONS=',I5,4X,
344                 1'WRONG=',I5,/,
345                 1     1X,'NUMBER OF CORRECT APC   RECOGNITIONS=',I5,4X,
346                 2'WRONG=',I5,/,
347                 2     1X,'NUMBER OF CORRECT TRUCK RECOGNITIONS=',I5,4X,
348                 3'WRONG=',I5,/,
349                 3     1X,'NUMBER OF CORRECT JEEP  RECOGNITIONS=',I5,4X,
350                 4'WRONG=',I5,//,
351                 4     1X,'NUMBER OF CORRECT M60   IDENTIFICATIONS=',I5,1X,
352                 4'WRONG=',I5,/,
353                 5     1X,'NUMBER OF CORRECT M48   IDENTIFICATIONS=',I5,1X,
354                 6'WRONG=',I5,/,
355                 6     1X,'NUMBER OF CORRECT M551  IDENTIFICATIONS=',I5,1X,
356                 6'WRONG=',I5,/,
357                 7     1X,'NUMBER OF CORRECT M113  IDENTIFICATIONS=',I5,1X,
358                 7'WRONG=',I5,/,
359                 8     1X,'NUMBER OF CORRECT LANCE IDENTIFICATIONS=',I5,1X,
360                 8'WRONG=',I5,/,
361                 9     1X,'NUMBER OF RESETS=',I8,/,
362                 1     1X,'NUMBER OF OPERATORS=',I8,/,
363                 1     1X,'NUMBER OF TOTAL OPERATOR RESPONSES=',I10)
364                 WRITE(6,47676)FRECOG,FALSEID,FRECOGF,FALSEIDF
365           47676 FORMAT(1X,
366                 1'NUMBER OF WRONG TARGET RECOGNITIONS    ON ACTUAL TARGETS=',
367                 1I10,/,
368                 2     1X,
369                 1'NUMBER OF WRONG TARGET IDENTIFICATIONS ON ACTUAL TARGETS='
370                 3I10,/,
371                 4     1X,'NUMBER OF FALSE TARGET RECOGNITIONS   =',I10,/,
372                 5     1X,'NUMBER OF FALSE TARGET IDENTIFICATIONS=',I10)
373                 WRITE(6,17676)IFSEQ4,IFSEQ13,IFSEQ17,(II,SEQS(II),II=1,18),SEQS(19)
374           17676 FORMAT(1X,' DESIGNATIONS FOR FALSE TARGET SEQUENCE  4=',I5,/,
375                 1     1X,' DESIGNATIONS FOR     NO TARGET SEQUENCE 13=',I5,/,
376                 2     1X,' DESIGNATIONS FOR     NO TARGET SEQUENCE 17=',I5,
377                 3     18(/,1X,'    SEQUENCE ',I2,' DESIGNATION COUNT=',I10),
378                 4       /,1X,'OUTSIDE GROUND TRUTH IRIG=',I10)
379                 IF(IDCOUNT.EQ.0)THEN
380                 TYPE *,' NO DETECTIONS'
381                 GO TO 1666
382                 ENDIF
383                 DELTADD=DELTAD/FLOAT(IDCOUNT)
384                 IF(IDCOUNT.EQ.1)THEN
385                 WRITE(6,912)DELTAD
```

```
912     FORMAT(1X,'DETECT TIME FOR THE SINGLE DETECTION=',F7.3)
        GO TO 1556
        ENDIF
        SD=SQRT((DELTADS-(FLOAT(IDCOUNT)*DELTADD**2))/(FLOAT(IDCOUNT)-1.0))
        WRITE(6,555)DELTADD,IDCOUNT,SD
555     FORMAT(1X,'MEAN DETECT TIME=',F7.3,1X,'SECONDS',1X,
        1'ON',I5,' DETECTIONS',/,1X,'WITH A STANDARD DEVIATION OF',
        2F7.3)
1556    IF(IRCOUNT.EQ.0)THEN
        TYPE *,' NO RECOGNITIONS'
        GO TO 1557
        ENDIF
        DELTARR=DELTAR/FLOAT(IRCOUNT)
        IF(IRCOUNT.EQ.1)THEN
        WRITE(6,913)DELTAR
913     FORMAT(1X,'RECOGNITION TIME FOR SINGLE RECOGNITION=',F7.3)
        GO TO 1557
        ENDIF
        SD=SQRT((DELTARS-(FLOAT(IRCOUNT)*DELTARR**2))/(FLOAT(IRCOUNT)-1.0))
        WRITE(6,556)DELTARR,IRCOUNT,SD
556     FORMAT(1X,'MEAN RECOGNITION TIME=',F10.3,1X,'SECONDS',1X,
        1'ON',I7,' TARGETS',/,1X,'WITH A STANDARD DEVIATION OF',
        2F10.3)
1557    IF(IICOUNT.EQ.0)THEN
        TYPE *,' NO IDENTIFICATIONS.'
        GO TO 1558
        ENDIF
        DELTAII=DELTAI/FLOAT(IICOUNT)
        IF(IICOUNT.EQ.1)THEN
        WRITE(6,914)DELTAI
914     FORMAT(1X,'IDENTIFICATION TIME FOR SINGLE ID=',F7.3)
        GO TO 1558
        ENDIF
        SD=SQRT((DELTAIS-(FLOAT(IICOUNT)*DELTAII**2))/(FLOAT(IICOUNT)-1.0))
        WRITE(6,557)DELTAII,IICOUNT,SD
557     FORMAT(1X,'MEAN IDENTIFICATION TIME=',F10.3,1X,'SECONDS',1X,
        1'ON',I7,' TARGETS',/,1X,'WITH A STANDARD DEVIATION OF',
        2F10.3)
1558    CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C       1. OBJECT DESIGNATION TIMES
C       2. TARGET DETECTION AND DESIGNATION TIMES
C       3. TARGET RECOGNITION TIMES
C       4. TARGET IDENTIFICATION TIMES
C
C       1,2,3 AND 4 ARE ALL CORRELATED WITH MINI RANGER IRIGS TO CALCULATE
C       RANGES. GROUND TRUTH TELLS US WHICH TARGET TO CALCULATE THE RANGE
C       TOO. THE  VARIABLE ITANKI CONTAINS THE ACTUAL TARGET NUMBER THAT
C       WAS DETECTED.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        DO 3333 IPLOT=1,3
                DO INITALTS=1,4
                ALTS(INITALTS)=0
```

```
                        ENDDO
                IHCOUNT=0
CCCCCCCCCCCCCCCCCCCCCCCCCCCCC
                IF(IPLOT.EQ.1)IENDD=IHDELTAD
                IF(IPLOT.EQ.2)IENDD=IHDELTAR
                IF(IPLOT.EQ.3)IENDD=IHDELTAI
C               IENDD=IHDELTAR      !
CCCCCCCCCCCCCCCCCCCCCCCCCCCCC
                DO 6 I=1,IENDD
CCCCCCCCCCCCCCCCCCCCCCCCCCCCC
                IF(IPLOT.EQ.1)THEN
                STIME=HDELTAD(I)
                ITANKI=ITANKID(I)
C               TYPE*,'DETECTION PLOT ITANKI=',ITANKI
                ENDIF
                IF(IPLOT.EQ.2)THEN
                STIME=HDELTAR(I)
                ITANKI=ITANKIR(I)
C               TYPE*,'RECOG PLOT ITANKI=',ITANKI
                ENDIF
                IF(IPLOT.EQ.3)THEN
                STIME=HDELTAI(I)
                ITANKI=ITANKII(I)
C               TYPE*,'ID PLOT ITANKI=',ITANKI
                ENDIF
C               STIME=HDELTAR(I)    !
CCCCCCCCCCCCCCCCCCCCCCCCCCCCC
                DO 5 J=1,IL
                TIHJ=TIH(J)
                TIMJ=TIM(J)
                TIMEMINI=FLOAT(TIDAY(J)-215)*60.*60.*60.+
                1TIHJ*60.*60.+TIMJ*60.+TSEC(J)
                IF(TIMEMINI.LT.STIME)GO TO 5
                IF(TIMEMINI.GT.STIME+8.)GO TO 7722
C               TYPE*,' TIMEMINI=',TIMEMINI,' STIME=',STIME,' DIFF=',
C               1ABS(STIME-TIMEMINI)
                IF(J.GT.1)THEN
                        ITIHH=TIH(J-1)
                        ITIMM=TIM(J-1)
                        ITDAY2=TIDAY(J-1)
                        SMTIME=TCVT(ITDAY2,ITIHH,ITIMM,TSEC(J-1))
C                       TYPE *,'SMTIME-STIME TESTING',SMTIME-STIME,'SMTIME=',
C       1               SMTIME,'STIME=',STIME
                                IF(ABS(SMTIME-STIME).LT.ABS(TIMEMINI-STIME))THEN
                        IF(ABS(SMTIME-STIME).GT.ERRORMINI)GO TO 7722
                                JJ=J-1
                                GO TO 7
                                ENDIF
                ENDIF
52              ITIHH=TIH(J+1)
                ITIMM=TIM(J+1)
                ITDAY2=TIDAY(J+1)
                SSTIME=TCVT(ITDAY2,ITIHH,ITIMM,TSEC(J+1))
C               TYPE *,' TIMEMINI-STIME TESTING',TIMEMINI-STIME
                IF(ABS(TIMEMINI-STIME).GE.ABS(STIME-SSTIME))GO TO 5
```

```
496                 IF(ABS(TIMEMINI-STIME).GT.ERRORMINI)GO TO 6
497           C     ERRORMINI IS USED AS A MAX ERROR ALLOWED FOR USABLE MINI RANGER INFORMATION
498                 JJ=J
499           7     TIMJ=TIM(JJ)
500                 TIMJ=TIM(JJ)
501                 TIMEMINI=FLOAT(TIDAY(JJ)-215)*60.*60.*60.+
502                 1TIMJ*60.*60.+TIMJ*60.+TSEC(JJ)
503                 TERROR=ABS(TIMEMINI-STIME)
504                 IF(TERROR.GE.ERRORMINI)GO TO 7722
505           C     TYPE*,'JJ=',JJ,'TX(JJ)=',TX(JJ),'VALID=',VALID,'ITANKI=',ITANKI,
506                 1'IRIG=',TIM(JJ),TIM(JJ),TSEC(JJ)
507           C     TYPE*,'YTANK=',YTANK(ITANKI),'IL=',IL,
508                 1'DELX=',DELX,'DELY=',DELY,'XTANK=',XTANK(ITANKI)
509
510                 DELX=-XTANK(ITANKI)+TX(JJ)
511                 DELY=-YTANK(ITANKI)+TY(JJ)
512                 DELZ=625.*.3048 !DEFAULT ALTITUDE FOR FALSE TARGET RUNS ETC.
513
514           C     TYPE *,' I=',I,' STIME=',STIME,'TIMEMINI=',TIMEMINI,' SSTIME=',
515           C     1SSTIME,' SMTIME=',SMTIME,'TX(JJ)=',TX(JJ),'TY(JJ)=',TY(JJ),
516           C     2'JJ=',JJ,'ITANKI=',ITANKI,'DELX=',DELX,'DELY=',DELY
517                 IDA=1
518           C SEQ1
519                 IF(STIME.GE.TCVT(215,18,55,12.)
520                 1.AND.STIME.LE.TCVT(215,18,56,12.138))THEN
521                 DELZ=.3048*825.
522           431   ALTS(3)=ALTS(3)+1
523                 IDA=5
524                 GO TO 1999
525                 ENDIF
526           C SEQ2
527                 IF(STIME.GE.TCVT(216,15,25,34.)
528                 1.AND.STIME.LE.TCVT(216,15,26,44.858))THEN
529                 DELZ=.3048*335.
530           C     GO TO 6          !********* FOR SELECTIVE ALTITUDE PLOTS
531           430   IDA=5
532                 ALTS(1)=ALTS(1)+1
533                 GO TO 1999
534                 ENDIF
535           C SEQ3
536                 IF(STIME.GE.TCVT(216,15,40,50.)
537                 1.AND.STIME.LE.TCVT(216,15,41,30.985))THEN
538                 DELZ=.3048*500.
539           C     GO TO 6          !********* FOR SELECTIVE ALTITUDE PLOTS
540           432   IDA=5
541                 ALTS(2)=ALTS(2)+1
542                 GO TO 1999
543                 ENDIF
544           C
545           C     NOTE THAT NOTHING IS HERE FOR SEQ 4 SINCE THEY SHOULD ALL BE INVALID
546           C
547           C SEQ5
548                 IF(STIME.GE.TCVT(215,19,7,6.)
549                 1.AND.STIME.LE.TCVT(215,19,8,22.218))THEN
550                 DELZ=.3048*500.
```

```
C       GO TO 6          !********* FOR SELECTIVE ALTITUDE PLOTS
433     IDA=0
        ALTS(2)=ALTS(2)+1
        GO TO 1999
        ENDIF
C SEQ6
        IF(STIME.GE.TCVT(216,16,1,30.)
        1.AND.STIME.LE.TCVT(216,16,2,27.432))THEN
        DELZ=.3048*825.
434     IDA=0
        ALTS(3)=ALTS(3)+1
        GO TO 1999
        ENDIF
C SEQ7
        IF(STIME.GE.TCVT(216,15,0,57.)
        1.AND.STIME.LE.TCVT(216,15,10,2.017))THEN
        DELZ=.3048*330.
C       GO TO 6 !********** FOR SELECTION OF SPECIFC ALTITUDES FOR PLOTTING
429     IDA=0
        ALTS(1)=ALTS(1)+1
        GO TO 1999
        ENDIF
C SEQ8
        IF(STIME.GE.TCVT(216,19,31,13.)
        1.AND.STIME.LE.TCVT(216,19,32,30.))THEN
        DELZ=325.*.3048
C       GO TO 6 !********** FOR SELECTION OF SPECIFC ALTITUDES FOR PLOTTING
428     IDA=0
        ALTS(1)=ALTS(1)+1
        GO TO 1999
        ENDIF
C SEQ9
        IF(STIME.GE.TCVT(216,15,50,52.)
        1.AND.STIME.LE.TCVT(216,15,51,52.322))THEN
        DELZ=.3048*500.
C       GO TO 6          !********* FOR SELECTIVE ALTITUDE PLOTS
435     IDA=0
        ALTS(2)=ALTS(2)+1
        GO TO 1999
        ENDIF
C SEQ10
        IF(STIME.GE.TCVT(216,15,58,0.)
        1.AND.STIME.LE.TCVT(216,15,59,2.229))THEN
        DELZ=.3048*825.
436     IDA=0
        ALTS(3)=ALTS(3)+1
        GO TO 1999
        ENDIF
C SEQ11
        IF(STIME.GE.TCVT(216,18,51,4.)
        1.AND.STIME.LE.TCVT(216,18,52,10.500))THEN
        DELZ=.3048*825.
437     IDA=0
        ALTS(3)=ALTS(3)+1
        GO TO 1999
```

```
      ENDIF
C SEQ12
      IF(STIME.GE.TCVT(216,15,44,10.)
     1.AND.STIME.LT.TCVT(216,15,45,0.))THEN
      DELZ=.3048*500.
C     GO TO 6          !********* FOR SELECTIVE ALTITUDE PLOTS
438   IDA=0
      ALTS(2)=ALTS(2)+1
      GO TO 1999
      ENDIF
C
C     NOTE THAT NOTHING IS HERE FOR SEQ 13 SINCE ALL SHOULD BE INVALID
C
C SEQ14
      IF(STIME.GE.TCVT(215,19,23,17.)
     1.AND.STIME.LE.TCVT(215,19,24,25.592))THEN
      DELZ=.3048*325.
C     GO TO 6 !********** FOR SELECTION OF SPECIFC ALTITUDES FOR PLOTTING
427   IDA=0
      ALTS(1)=ALTS(1)+1
      GO TO 1999
      ENDIF
C SEQ15
      IF(STIME.GE.TCVT(216,16,05,4.)
     1.AND.STIME.LE.TCVT(216,16,5,59.926))THEN
      DELZ=.3048*030.
439   IDA=0
      ALTS(3)=ALTS(3)+1
      GO TO 1999
      ENDIF
C     SEQ16 IS THE SAME AS 8 SO WE DON'T WANT TO COUNT THEM TWICE
C     IF(TIMEMINI.GE.TCVT(215,19,31,13.)
C    1.AND.TIMEMINI.LE.TCVT(215,19,32,30.))THEN
C     DELZ=.3048*325.
C     IDA=0
C     ALTS(1)=ALTS(1)+1
C     ENDIF
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     NOTE THAT NOTHING IS HERE FOR SEQ 17 SINCE ALL SHOULD BE INVALID
C
C SEQ18
      IF(STIME.GE.TCVT(216,15,22,25.)
     1.AND.STIME.LE.TCVT(216,15,23,8.361))THEN
      DELZ=.3048*330.
C     GO TO 6 !********** FOR SELECTION OF SPECIFC ALTITUDES FOR PLOTTING
426   IDA=0
      ALTS(1)=ALTS(1)+1
      GO TO 1999
      ENDIF
      IF(IDA.EQ.1)THEN
             ALTS(4)=ALTS(4)+1
      CALL CVTT(STIME,IDAYC,IHC,IMC,SECC)
      PRINT 1987,IDAYC,IHC,IMC,SECC,IENDD,I,STIME
1987  FORMAT(1X,'****OUTSIDE ALTITUDE FOR IRIG          ',
     11X,I3,':',I2,':',I2,':',F6.2,5X,'INDELTAX=',I6,
```

```
661                 25X,'CURRENT LIST INDEX=',I6,' STIME=',F10.2)
662                       GO TO 6
663                 ENDIF
664           1999  CONTINUE
665           C     DELZ=TZ(J)-172  !DOUG KWHITE TOLD ME TO TAKE OFF FOR SEA LEVEL
666                 RA=SQRT(DELX*DELX+DELY*DELY+DELZ*DELZ)
667                 IHCOUNT=IHCOUNT+1
668           CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
669                 IF(IPLOT.EQ.1)HDELTAD(IHCOUNT)=RA
670           C     IF(IPLOT.EQ.1)TYPE *,'STIME,DAY,HR,MIN,SEC,IHCOUNT=',
671           C     1STIME,IDAY,IHH,IRR,IMM,SEC,IHCOUNT
672                 IF(IPLOT.EQ.2)HDELTAR(IHCOUNT)=RA
673                 IF(IPLOT.EQ.3)HDELTAI(IHCOUNT)=RA
674           C     HDELTAR(IHCOUNT)=RA
675           CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
676                 GO TO 6
677           5     CONTINUE
678           7722  CALL CVTT(STIME,IDAYC,IHC,IMC,SECC)
679                 PRINT 987,IDAYC,IHC,IMC,SECC,IENDD,I,STIME
680           987   FORMAT(1X,'** MINI RANGER DATA NOT FOUND FOR IRIG',
681                 11X,I3,':',I2,':',I2,':',F6.2,5X,'IHDELTAX=',I6,
682                 25X,'CURRENT LIST INDEX=',I6,' STIME=',F10.2)
683           6     CONTINUE
684                 IF(IPLOT.EQ.1)THEN
685           C     TYPE *,'IHCOUNT=',IHCOUNT,' IPLOT=',IPLOT
686                 IHCOUNTD=IHCOUNT
687                 WRITE(6,27676)(ALTT(II),ALTS(II),II=1,3),ALTS(4)
688           27676 FORMAT(3(1X,'NUMBER OF DETECTIONS AT ',I3,' FEET =',I10,/)
689                 11X,'NUMBER OF DETECTIONS OUT OF THE ABOVE ALTS=',I10)
690                 ENDIF
691                 IF(IPLOT.EQ.2)THEN
692           C     TYPE *,'IHCOUNT=',IHCOUNT,' IPLOT=',IPLOT
693                 IHCOUNTR=IHCOUNT
694                 WRITE(6,27675)(ALTT(II),ALTS(II),II=1,3),ALTS(4)
695           27675 FORMAT(3(1X,'NUMBER OF RECOGNITIONS AT ',I3,' FEET =',I10,/)
696                 11X,'NUMBER OF RECOGNITIONS OUT OF THE ABOVE ALTS=',I10)
697                 ENDIF
698                 IF(IPLOT.EQ.3)THEN
699                 IHCOUNTI=IHCOUNT
700           C     TYPE *,'IHCOUNT=',IHCOUNT,' IPLOT=',IPLOT
701                 WRITE(6,27674)(ALTT(II),ALTS(II),II=1,3),ALTS(4)
702           27674 FORMAT(3(1X,'NUMBER OF IDENTIFICATIONS AT ',I3,' FEET =',I10,/)
703                 11X,'NUMBER OF IDENTIFICATIONS OUT OF THE ABOVE ALTS=',I10)
704                 ENDIF
705                 TYPE *,' ERROR ALLOWABLE IN MINIRANGER DATA =',ERRORMINI
706           3333  CONTINUE
707           C     CALL HIST2(1,HDELTAO,IHCOUNT)   ! FOR OBJECT RANGE DESIGNATIONS
708           C     DO IPRINT=1,IHCOUNTD
709           C     TYPE *,HDELTAD(IPRINT),IPRINT,IHCOUNTD
710           C     ENDDO
711                 CALL HIST2(2,HDELTAD,IHCOUNTD)  ! FOR TARGET DETECTION RANGE&DESIGNATIONS
712           C     DO IPRINT=1,IHCOUNTR
713           C     TYPE *,HDELTAR(IPRINT),IPRINT,IHCOUNTR
714           C     ENDDO
715                 CALL HIST2(3,HDELTAR,IHCOUNTR)  ! FOR RECOGNITION DELTA HISTOGRAM
```

```
716           C       DO IPRINT=1,IHCOUNTI
717           C       TYPE *,HDELTAI(IPRINT),IPRINT,IHCOUNTI
718           C       ENDDO
719                   CALL HIST2(4,HDELTAI,IHCOUNTI)  ! FOR IDENTIFICATION DELTA HISTOGRAM
720                   STOP ' ALL DONE'
721           7676    CONTINUE
722           C       TYPE *,' TAPE NUM LINE READ'
723                   NOPER=NOPER+1
724                   ITANKI=0
725                   VALID=0
726                   GO TO 1
727                   END
728                   FUNCTION TCVT(IDAY,IH,IM,SEC)
729                   TCVT=FLOAT(IDAY-215)*60.*60.*60.+FLOAT(IH)*60.*60.+
730                  1FLOAT(IM)*60.+SEC
731                   RETURN
732                   END
733                   SUBROUTINE GTRUTH(IDAY,IH,IM,SEC,X,Y,VALID,ITARGETN)
734                   PARAMETER MEM=30000
735                   BYTE IH,IM,VALID,TIH(MEM),TIM(MEM),SEQN,TARGETS(10,18)
736                   BYTE NUMT(18)
737                   DIMENSION TSEC(MEM),ERRORP(18)
738                   INTEGER*2 TIDAY(MEM),BOX(4,MEM),IDAY,BOXES(4),BOXN,T(13),SEQS(19)
739                   INTEGER TDAY,TIHH,TIMM,ITARGETN,TDAY2,TIHH2,TIMM2,BEGINS,ENDS
740                   DATA GERROR/10./
741           C       MAX ALLOWABLE SKIP IN GROUND TRUTH
742           C       NUMT(X) CONTAINS THE NUMBER OF TARGETS TRACKED DURING THE X RUN SEQ
743                   DATA NUMT/6,7,4,0,4,9,6,3,8,9,8,4,0,4,4,3,0,2/
744           C       TARGETS IS A BYTE ARRAY THAT CONVERTS T NUMBERS TO ACTUAL TARGET NUMBERS
745           C       THE FIRST INDEX IS THE T NUMBER 1 TO 10
746           C       THE SECOND INDEX IS THE SEQUENCE NUMBER 1 TO 18 ON TAPE 1
747           C       SEQS(19) IS THE COUNT FOR ALL OUT OF IRIG GROUND TRUTH DESIGNATIONS
748                   COMMON/COUNTS/IFSEQ4,IFSEQ13,IFSEQ17,SEQS
749                   DATA SEQS/19*0/
750                   DATA INIT/1/,IFSEQ4/0/,IFSEQ13/0/,IFSEQ17/0/
751                   DATA ((TARGETS(N,M),N=1,10),M=1,18)/4,2,1,9,8,7,4*1,
752                   29,8,6,7,4,10,1,3*1,
753                   37,8,9,10,6*1,
754                   41*1,
755                   51,2,4,3,6*1,
756                   69,8,7,6,5,4,3,13,10,1,
757                   73,9,8,5,7,6,4*1,
758                   81,2,4,7*1,
759                   910,1,4,5,6,7,8,9,2*1,
760                   113,1,12,4,5,6,7,8,9,1,
761                   19,8,7,6,4,3,2,1,2*1,
762                   27,8,9,10,6*1,
763                   310*1,
764                   41,2,3,4,6*1,
765                   510,12,13,1,6*1,
766                   61,2,4,7*1,
767                   710*1,
768                   813,1,8*1/
769                   IF(INIT)THEN
770                   TYPE *,' READING GROUND TRUTH DATA FILE GROUND.FFF.. STANDBY.'
```

```
771                     OPEN(UNIT=9,NAME='GROUND.FFF',TYPE='OLD',READONLY,SHARED)
772                     IC=0
773             2       READ(9,221,END=100)IDAY,IH,IM,SEC,(BOXES(I),I=1,4)
774             221     FORMAT(1X,Z4,1X,I2,1X,I2,1X,F7.3,4(3X,I4))
775                     IC=IC+1
776                     BOX(1,IC)=BOXES(1)      !LEFT
777                     BOX(2,IC)=BOXES(2)      !RIGHT
778                     BOX(3,IC)=BOXES(3)      !TOP
779                     BOX(4,IC)=BOXES(4)      !BOTTOM
780                     TIH(IC)=IH
781                     TIM(IC)=IM
782                     TSEC(IC)=SEC
783                     TIDAY(IC)=IDAY
784             C       WRITE(6,67)IH,IM,SEC
785             67      FORMAT(1X,I2,':',I2,':',F6.2)
786                     GO TO 2
787             100     TYPE *,'TOTAL NUMBER OF IRIGS=',IC
788                     INIT=0
789                     RETURN
790                     ENDIF
791             C.....................................................................
792             C       ENTER HERE AFTER INITIAL GROUND TRUTH FILE READ
793             C.....................................................................
794                     VALID=0
795                     IHR=IH
796                     IMR=IM
797                     IDDAY=IDAY
798                     TIME=TCVT(IDDAY,IHR,IMR,SEC)
799             C       PRINT 16,IDAY,IH,IM,SEC,TIME
800             16      FORMAT(1X,' TIME=',I3,':',I2,':',I2,':',F6.2,5X,F20.5)
801             C       SEQ 1
802                     IF(TIME.GE.TCVT(215,10,55,21.000).AND.
803                     1  TIME.LE.TCVT(215,10,56,12.130))THEN
804             C       RETURN  ! ************SELECTIVE SEQUENCE PROCESSING
805             800     BEGINS=20009
806                     ENDS=3233
807                     SEQN=1
808                     SEQS(1)=SEQS(1)+1
809                     GO TO 1
810                     ENDIF
811             C       SEQ 2
812                     IF(TIME.GE.TCVT(216,15,25,59.455).AND.
813                     1  TIME.LE.TCVT(216,15,26,44.850))THEN
814             C       RETURN  ! ************SELECTIVE SEQUENCE PROCESSING
815             900     BEGINS=60700
816                     ENDS=7157
817                     SEQN=2
818                     SEQS(2)=SEQS(2)+1
819                     GO TO 1
820                     ENDIF
821             C       SEQ3
822                     IF(TIME.GE.TCVT(216,15,41, 6.100).AND.
823                     1  TIME.LE.TCVT(216,15,41,30.985))THEN
824             C       RETURN  ! ************SELECTIVE SEQUENCE PROCESSING
825             807     SEQN=3
```

```
826                 BEGINS=7150
827                 ENDS=7602
828                 SEQS(3)=SEQS(3)+1
829                 GO TO 1
830                 ENDIF
831           C     SEQ4    FALSE TARGET RUN
832                 IF(TIME.GE.TCVT(216,15,34,12.000).AND.
833                1  TIME.LE.TCVT(216,15,35,22.000))THEN
834                 IFSEQ4=IFSEQ4+1
835           C     TYPE *,' FALSE TARGET IN RUN SEQ 17',IFSEQ4
836                 RETURN
837                 ENDIF
838           C     SEQ5
839                 IF(TIME.GE.TCVT(215,19, 7,45.232).AND.
840                1  TIME.LE.TCVT(215,19, 8,22.210))THEN
841           C     RETURN  | ************SELECTIVE SEQUENCE PROCESSING
842           886   SEQN=5
843                 BEGINS=3234
844                 SEQS(5)=SEQS(5)+1
845                 ENDS=4141
846                 GO TO 1
847                 ENDIF
848           C     SEQ6
849                 IF(TIME.GE.TCVT(216,16, 1,38.918).AND.
850                1  TIME.LE.TCVT(216,16, 2,27.432))THEN
851           C     RETURN  | ************SELECTIVE SEQUENCE PROCESSING
852           793   SEQN=6
853                 BEGINS=10153
854                 ENDS=11575
855                 SEQS(6)=SEQS(6)+1
856                 GO TO 1
857                 ENDIF
858           C     SEQ7
859                 IF(TIME.GE.TCVT(215,15, 9, 1.026).AND.
860                1  TIME.LE.TCVT(215,15,10, 2.017))THEN
861           C     RETURN  | ************SELECTIVE SEQUENCE PROCESSING
862           885   SEQN=7
863                 BEGINS=1
864                 ENDS=756
865                 SEQS(7)=SEQS(7)+1
866                 GO TO 1
867                 ENDIF
868           C     SEQ8
869                 IF(TIME.GE.TCVT(215,19,32, 2.424).AND.
870                1  TIME.LE.TCVT(215,19,32,29.301))THEN
871           C     RETURN  | ************SELECTIVE SEQUENCE PROCESSING
872           884   SEQN=8
873                 BEGINS=4803
874                 ENDS=5590
875                 SEQS(8)=SEQS(8)+1
876                 GO TO 1
877                 ENDIF
878           C     SEQ9
879                 IF(TIME.GE.TCVT(216,15,51, 9.556).AND.
880                1  TIME.LE.TCVT(216,15,51,52.322))THEN
```

```
                SEQN=9
      C         RETURN  ! ************SELECTIVE SEQUENCE PROCESSING
      803       BEGINS=8191
                ENDS=9124
                SEQS(9)=SEQS(9)+1
                GO TO 1
                ENDIF
      C         SEQ 10
                IF(TIME.GE.TCVT(216,15,58, 5.345).AND.
                1  TIME.LE.TCVT(216,15,59, 2.229))THEN
      C         RETURN  ! ************SELECTIVE SEQUENCE PROCESSING
      802       SEQN=10
                BEGINS=9125
                ENDS=10152
                SEQS(10)=SEQS(10)+1
                GO TO 1
                ENDIF
      C         SEQ 11
                IF(TIME.GE.TCVT(216,18,51,17.250).AND.
                1  TIME.LE.TCVT(216,18,52,10.500))THEN
      C         RETURN  ! ************SELECTIVE SEQUENCE PROCESSING
      801       SEQN=11
                BEGINS=757
                ENDS=2000
                SEQS(11)=SEQS(11)+1
                GO TO 1
                ENDIF
      C         SEQ12
                IF(TIME.GE.TCVT(216,15,44,28.438).AND.
                1  TIME.LE.TCVT(216,15,44,57.563))THEN
      C         RETURN  ! ************SELECTIVE SEQUENCE PROCESSING
      800       SEQN=12
                BEGINS=7603
                ENDS=8190
                SEQS(12)=SEQS(12)+1
                GO TO 1
                ENDIF
      C         SEQ 13 FALSE TARGET RUN
                IF(TIME.GE.TCVT(216,15,49,30.000).AND.
                1  TIME.LE.TCVT(216,15,50,19.000))THEN
                IFSEQ13=IFSEQ13+1
      C         TYPE *,' FALSE TARGET IN RUN SEQ 13',IFSEQ13
                RETURN
                ENDIF
      C         SEQ 14
                IF(TIME.GE.TCVT(216,19,23,57.769).AND.
                1  TIME.LE.TCVT(216,19,24,25.592))THEN
      C         RETURN  ! ************SELECTIVE SEQUENCE PROCESSING
      799       SEQN=14
                BEGINS=4142
                ENDS=4802
                SEQS(14)=SEQS(14)+1
                GO TO 1
                ENDIF
      C         SEQ 15
```

```
936                     IF(TIME.GE.TCVT(216,16, 5,21.414).AND.
937                     1  TIME.LE.TCVT(216,16, 5,59.926))THEN
938             C       RETURN  ! ************SELECTIVE SEQUENCE PROCESSING
939             798     SEQN=15
940                     BEGINS=11576
941                     ENDS=11861
942                     SEQS(15)=SEQS(15)+1
943                     GO TO 1
944                     ENDIF
945             C       SEQ 16
946                     IF(TIME.GE.TCVT(215,19,32, 2.424).AND.
947                     1  TIME.LE.TCVT(215,19,32,29.576))THEN
948             C       RETURN  ! ************SELECTIVE SEQUENCE PROCESSING
949             797     SEQN=16
950                     BEGINS=4803
951                     ENDS=559
952                     SEQS(16)=SEQS(16)+1
953                     GO TO 1
954                     ENDIF
955             C       SEQ 17 FALSE TARGET RUN
956                     IF(TIME.GE.TCVT(215,19,34, 4.000).AND.
957                     1  TIME.LE.TCVT(215,19,34,39.000))THEN
958                     IFSEQ17=IFSEQ17+1
959             C       TYPE *,' FALSE TARGET IN RUN SEQ 17',IFSEQ17
960                     RETURN
961                     ENDIF
962             C       SEQ 18
963                     IF(TIME.GE.TCVT(216,15,22,35.480).AND.
964                     1  TIME.LE.TCVT(216,15,23, 8.116))THEN
965             C       RETURN  ! ************SELECTIVE SEQUENCE PROCESSING
966             796     SEQN=18
967                     BEGINS=5591
968                     ENDS=6069
969                     SEQS(18)=SEQS(18)+1
970                     GO TO 1
971                     ENDIF
972                     ITDAY=IAND(IDAY,'00FF'X)
973                     SEQS(19)=SEQS(19)+1
974             C       WRITE(45,62)ITDAY,IH,IM,SEC
975             62      FORMAT(1X,'******* NO SEQUENCE IRIG FOR ',I3,':',I2,':',I2,':',F6.2)
976                     RETURN
977             1       DO I=1,NUMT(SEQN)
978                     T(I)=0
979                     ENDDO
980             C<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
981             C
982             C       CHECK TO SEE IF X,Y DESIGNATION IS ON THE MENU
983             C
984             C>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
985                     IF(Y.GT.254.)THEN
986             C       HE'S IN THE MENU.....LET'S GO SEE IF HE GOT IT RIGHT.
987             C
988             C       FROM THIS IRIG LAST DETECTION WAS ITARGETN
989                     VALID=3
990                     RETURN
```

```
      ENDIF
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     HE TOUCHED SOMETHING ON THE SCREEN.. NOT THE MENU.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      ITARGETN=0       !RESET THE TARGET NUMBER FOR NEXT MENU INPUT CHECK
C     TYPE *,' OPERATOR INPUT.....',X,Y
      DO 5 I=1,IC
      TDAY=IAND(TIDAY(I),'00FF'X)
      TIHH=TIH(I)
      TIMM=TIM(I)
      GTIME=TCVT(TDAY,TIHH,TIMM,TSEC(I))
C     PRINT 156,TDAY,TIHH,TIMM,TSEC(I),GTIME,I
156   FORMAT(1X,'GTIME=',I3,':',I2,':',I2,':',F6.2,5X,F20.5,5X,I6)
      IF(GTIME.LT.TIME)GO TO 5
      TDAY2=IAND(TIDAY(I+1),'00FF'X)
      TIHH2=TIH(I+1)
      TIMM2=TIM(I+1)
      GGTIME=TCVT(TDAY2,TIHH2,TIMM2,TSEC(I+1))
      IF(GTIME.EQ.GGTIME)GO TO 5
      IF(ABS(GTIME-TIME).GT.GERROR)GO TO 5
C     TYPE *,' TIME=',TIME,' GTIME=',GTIME,' GGTIME=',GGTIME
C     TYPE *,' I=',I,' TIME=',TIME,' GTIME=',GTIME
      I1=I
      IF(I1.GT.BEGINS)I1=I-1
      I2=BEGINS
      I3=-1
      I4=0
101   IF(I.LT.BEGINS.OR.I.GT.ENDS)THEN
      TYPE *,' GROUND TRUTH INDEX INVALID'
      TYPE *,' BEGINS=',BEGINS,' ENDS=',ENDS,' I=',I,' SEQN=',SEQN
      RETURN
      ENDIF
C     IF(I4.EQ.0)TYPE *,' SEARCH DOWN IRIG'
C     IF(I4.EQ.1)TYPE *,' SEARCH UP   IRIG'
      DO 111 IA=I1,I2,I3
      IZBOX=0
      DO 900 JJJ=1,4
      IF(BOX(JJJ,IA).NE.0)GO TO 900
      IZBOX=IZBOX+1
900   CONTINUE
      IF(IZBOX.EQ.4)GO TO 111
      BOXN=ISHFT(TIDAY(IA),-12)
C     TYPE *,' BOXN=',BOXN
      IF(T(BOXN).NE.0)GO TO 111
      T(BOXN)=IA
      DO 112 ITN=1,NUMT(SEQN)
C     TYPE *,'T(ITN)=',T(ITN),' SEQN=',SEQN,' BOXN=',BOXN,
C    1' IA=',IA,'NO. TARGETS=',NUMT(SEQN),'ITN=',ITN
      IF(T(ITN).NE.0)THEN
      GO TO 112
      ENDIF
      GO TO 111
112   CONTINUE
```

```
1046                    GO TO 212
1047            111     CONTINUE
1048                    GO TO 214
1049            C       IF WE GET HERE ALL TARGET BOXES HAVE BEEN LOCATED
1050            C       FOR THIS OPERATOR IRIG SO LET'S COMPARE X,Y
1051            C212    TYPE *,'OPERATOR INPUT=',X,Y
1052            C       DO JJJ=1,NUMT(SEQN)
1053            C       L9=T(JJJ)
1054            C       TYPE *,' ',TIDAY(L9),':',TIH(L9),':',TIM(L9),':',TSEC(L9),
1055            C       1' ',BOX(1,L9),BOX(2,L9),BOX(3,L9),BOX(4,L9),
1056            C       2' TARGET #=',TARGETS(JJJ,SEQN)
1057            C       ENDDO
1058            C       ACCEPT *,JJJ
1059            212     CONTINUE
1060                    DO 213 ITN=1,NUMT(SEQN)
1061                    ERRORP(ITN)=99999.
1062                    ERRORX=.5*FLOAT(BOX(1,T(ITN))-BOX(2,T(ITN)))
1063                    ERRORY=.5*FLOAT(BOX(3,T(ITN))-BOX(4,T(ITN)))
1064                    IF(ERRORX.LT.15.)ERRORX=15.
1065                    IF(ERRORY.LT.15.)ERRORY=15.
1066            C       PRINT 1214,X,Y,(BOX(K,T(ITN)),K=1,4)
1067            1214    FORMAT(1X,'X=',F6.2,' Y=',F6.2,' BOX=',4(1X,I4))
1068                    XLEFT=FLOAT(BOX(1,T(ITN)))-ERRORX
1069                    XRIGHT=FLOAT(BOX(2,T(ITN)))+ERRORX
1070                    IF(X.LT.XLEFT.OR.
1071                    1  X.GT.XRIGHT)GO TO 213
1072                    YTOP=FLOAT(BOX(3,T(ITN)))-ERRORY
1073                    YBOTTOM=FLOAT(BOX(4,T(ITN)))+ERRORY
1074                    IF(Y.LT.YTOP.OR.
1075                    1  Y.GT.YBOTTOM)GO TO 213
1076                    CX=(XLEFT+XRIGHT)/2.0
1077                    CY=(YTOP+YBOTTOM)/2.0
1078                    XERR=ABS(X-CX)
1079                    YERR=ABS(Y-CY)
1080                    ERRORP(ITN)=SQRT(XERR**2+YERR**2)
1081            C       TYPE *,' WE HAVE A GOOD ONE HERE!!!!'
1082            C       WE HAVE A GOOD ONE HERE!!!!!
1083            213     CONTINUE
1084                    XMAXE=99999.
1085                    ITARG=0
1086                    DO ITN=1,NUMT(SEQN)
1087            C       TYPE *,' ITN=',ITN,' ERRORP(ITN)=',ERRORP(ITN)
1088                    IF(ERRORP(ITN).LT.XMAXE)THEN
1089                    ITARG=ITN
1090                    XMAXE=ERRORP(ITN)
1091                    ENDIF
1092                    ENDDO
1093                    IF(ITARG.EQ.0)THEN
1094            C       TYPE *,' OPERATOR DIDN''T HIT ANYWHERE NEAR A TARGET!!'
1095            C       TYPE *,'X=',X,' Y=',Y
1096                    RETURN  ! OPERATOR DIDN''T HIT ANYWHERE NEAR A TARGET!!
1097                    ENDIF
1098            222     VALID=1
1099                    ITARGETN=TARGETS(ITARG,SEQN)
1100            C       TYPE *,'X=',X,' Y=',Y,' GROUND TRUTH TARGET NUMBER=',ITARGETN
```

```
1101                    RETURN
1102            214     IF(I4.EQ.0)THEN
1103                    I1=I
1104                    I2=ENDS
1105                    I3=1
1106                    I4=1
1107                    GO TO 101
1108                    ENDIF
1109                    TYPE *,' * CAN''T FIND ALL TARGETS IN GROUND TRUTH SEARCH'
1110                    DO 215 IA=1,NUMT(SEQN)
1111                    PRINT 216,IA,T(IA)
1112            216     FORMAT(1X,'IA=',I2,' T(IA)=',I6)
1113            215     CONTINUE
1114                    TYPE *, 'WE REALLY HAVE PROBLEMS IF THIS IS PRINTED'
1115                    RETURN          !WE REALLY HAVE PROBLEMS IF WE RETURN FROM HERE...
1116            C44     TYPE *,' TIME>GTIME=',TIME,GTIME
1117            5       CONTINUE
1118            7722    ITDAY=IAND(IDAY,'00FF'X)
1119                    PRINT 6,ITDAY,IH,IM,SEC
1120            6       FORMAT(1X,'******** NO GROUND TRUTH FOR ',I3,':',I2,':',I2,':',F6.2)
1121                    RETURN
1122                    END
1123                    SUBROUTINE CVTT(STIME,IDAYC,IHC,IMC,SEC)
1124                    IF(STIME.GT.60.*60.*60.)THEN
1125                    IDAYC=216
1126                    TIME=STIME-(60.*60.*60.)
1127                    IHC=INT(TIME/(60.*60.))
1128                    TIME=TIME-(FLOAT(IHC)*60.*60.)
1129                    IMC=INT(TIME/60.)
1130                    SEC=TIME-(FLOAT(IMC)*60.)
1131                    RETURN
1132                    ENDIF
1133                    IDAYC=215
1134                    IHC=INT(STIME/(60.*60.))
1135                    TIME=STIME-(FLOAT(IHC)*60.*60.)
1136                    IMC=INT(TIME/60.)
1137                    SEC=TIME-(FLOAT(IMC)*60.)
1138                    RETURN
1139                    END
```

APPENDIX C

GROUND TRUTH DATA COLLECTION PROGRAM

```
C*****************************************************************************
C
C       IRIG(1)=FIRST IRIG TIME WORD
C       IRIG(2)=SECOND IRIG TIME WORD
C       IRIG(3)=THIRD IRIG TIME WORD
C       IRIG(4)=GATE SIZES OF THE TRACK GATE (TOP AND BOTTOM)
C       IRIG(5)=GATE SIZES OF THE TRACK GATE (LEFT AND RIGHT)
C       IRIG(6)=16 BITS THAT CONTAIN THE X AND Y POSITION OF THE CURSOR
C
C       BOXES(40) IS THE ARRAY THAT CONTAINS THE 10 BOX POSITIONS
C       LEFT,RIGHT,UP AND DOWN<-- IN THAT ORDER
C
C*****************************************************************************
        EXTERNAL IO$_READVBLK,IO$_WRITEVBLK,QXRA0
        DOUBLE PRECISION QUAD
        INTEGER SYS$WAITFR,SYS$BINTIM,TTCHAN
        BYTE C
        INTEGER*2 IRIG(6),IY(2),MD,TD,UD,TH,UH,TM,UM,TS1,TS2,US,LS,MS
        INTEGER*2 OBOX5,OBOX6,OBOX7,OBOX8,OBOX5M1,OBOX5P1,OBOX7M1
        INTEGER*2 OBOX7P1,IF1,TFLAG
        INTEGER*2 OBOX6M1,OBOX6P1,OBOX8M1,OBOX8P1,OPASS
        PARAMETER MEM1=10000
        PARAMETER NOISE=2
        BYTE IM,IH
        INTEGER*2 IDAY
        INTEGER*2 TINDEX,OIH
C       TINDEX IS THE ARRAY INDEX FOR THE TARGET INFORMATION
        BYTE TAPECNTRL
        INTEGER*2 IOSB(4),BOXES(40),BLINKM,ISAVE(2)
        INTEGER SYS$QIOW,SYS$ASSIGN,SYS$QIO,SYS$LKWSET
        INTEGER INLOCK(2),IOLOCK(2)
        PARAMETER TEN=10
        CHARACTER*16 TIME,TIMED(TEN)
        INTEGER*2 TTH(TEN),TUH(TEN),TTM(TEN),TUM(TEN),TTS1(TEN)
        INTEGER*2 TUS(TEN)
        INTEGER*2 PASS
        BYTE X,Y,XY(2),TOP,BOTTOM,LEFT,RIGHT,TB(2),LR(2)
        EQUIVALENCE(XY(1),X),(XY(2),Y),(IRIG(6),XY(1))
        EQUIVALENCE(TB(1),BOTTOM),(TB(2),TOP),(IRIG(5),TB(1))
        EQUIVALENCE(LR(1),LEFT),(LR(2),RIGHT),(IRIG(4),LR(1))
        EQUIVALENCE(IRIG2,IRIG(2))
        COMMON/CHAN/TTCHAN
        DATA N/1/
        DATA IF1/0/
        DATA TINDEX/0/,TFLAG/0/
        DATA  TTH/10*1/
        DATA  TUH/10*3/
        DATA  TTM/0,0,0,0,0,1,0,0,0,0/
        DATA  TUM/0,1,3,5,9,0,0,0,0,0/
        DATA TTS1/0,5,3,3,0,3,0,0,0,0/
        DATA  TUS/0,0,0,0,0,5,0,0,0,0/
        DATA TIMED/
        1'0000 00:00:17.00',
        2'0000 00:00:07.50',
        3'0000 00:00:05.00',
```

```
56                    3'$$$$ $$:$$:$6.5$',
57                    4'$$$$ $$:$$:16.$$',
58                    5'$$$$ $$:$$:$3.7$',
59                    64"'$$$$ $$:$$:$$.$$'/
6$                    PARAMETER STOP=1
61                    PARAMETER FF=2
62                    PARAMETER FORWARD=4
63                    PARAMETER REWIND=8
64                    ISTATUS=SYS$ASSIGN('XRA$',GRCHAN,,)
65                    IF(.NOT.ISTATUS)TYPE *,' ERROR IN GRA1 CHANNEL ASSIGN'
66                    ISTATUS=SYS$ASSIGN('TTB5',TTCHAN,,)
67                    IF(.NOT.ISTATUS)
68                    1TYPE *,' ERROR IN TTB5 ACOUSTIC TOUCH SCREEN CHANNEL ASSIGN'
69                    CALL QASTTB5
7$                    TYPE *,' ENTER CONTROL Y WHEN ALL DONE WITH THIS TAPE PASS'
71                    PASS=1
72                    IF(PAS9.EQ.1)THEN
73                    OPEN(UNIT=9,NAME='TARGETS1.DAT',TYPE='NEW',FORM='UNFORMATTED')
74                    ENDIF
75              C     TAPECNTRL=STOP
76              C     CALL TAPEREMOT(TAPECNTRL)
77              C     TAPECNTRL=FORWARD
78              C     CALL TAPEREMOT(TAPECNTRL)
79                    INLOCK(1)=%LOC(BOXES(1))
8$                    INLOCK(2)=%LOC(BOXES(4$))
81                    K=SYS$LKWSET(INLOCK,IOLOCK,)
82                    TYPE *,' INLOCK(1)= ',INLOCK(1),' INLOCK(2)= ',INLOCK(2)
83                    TYPE *,' IOLOCK(1)= ',IOLOCK(1),' IOLOCK(2)= ',IOLOCK(2)
84                    IF(.NOT.K)TYPE *,' UNABLE TO LOCK BOXES I/O BUFFER'
85                    INLOCK(1)=%LOC(IRIG(1))
86                    INLOCK(2)=%LOC(IRIG(6))
87                    K=SYS$LKWSET(INLOCK,IOLOCK,)
88                    TYPE *,' INLOCK(1)= ',INLOCK(1),' INLOCK(2)= ',INLOCK(2)
89                    TYPE *,' IOLOCK(1)= ',IOLOCK(1),' IOLOCK(2)= ',IOLOCK(2)
9$                    IF(.NOT.K)TYPE *,' UNABLE TO LOCK IRIG I/O BUFFER'
91              1     K=SYS$QIOW(%VAL(1),%VAL(GRCHAN),%VAL(%LOC(IO$_READVBLK)),IOSB,
92              C     1QXRA$,C,IRIG(1),%VAL(12),,,,)
93                    1,,IRIG(1),%VAL(12),,,,)
94                    IF(IRIG(3))GO TO 1
95              C     WRITE(6,567)X,Y,IRIG(1),IRIG(2),IRIG(3)
96              567   FORMAT(1X,'X=',O3,5X,'Y=',O3,1X,O6,1X,O6,1X,O6)
97                    HD=ISHFT(IRIG(1),-12)
98                    TD=IAND(ISHFT(IRIG(1),-8),'F'X)
99                    UD=IAND(ISHFT(IRIG(1),-4),'F'X)
1$$                   TH=IAND(IRIG(1),'3'X)
1$1                   UH=ISHFT(IRIG(2),-12)
1$2                   TH=IAND(ISHFT(IRIG(2),-9),'7'O)
1$3                   UM=IAND(ISHFT(IRIG(2),-8),'F'X)
1$4                   TS1=IAND(ISHFT(IRIG(2),-2),'7'O)
1$5                   ISAVE(1)=IAND(ISHFT(IRIG(2),2),'C'X)
1$6                   ISAVE(2)=ISHFT(IRIG(3),-14)
1$7                   US=IOR(ISAVE(1),ISAVE(2))
1$8                   TS2=IAND(ISHFT(IRIG(3),-1$),'F'X)
1$9                   LS=IAND(ISHFT(IRIG(3),-6),'F'X)
11$                   MS=IAND(ISHFT(IRIG(3),-2),'F'X)
```

```
111                 IRIG2=ISHFT(IRIG2,-2)
112           C     WRITE(6,560)MD,TD,UD,TH,UH,TM,UM,TS1,US,TS2,LS,MS
113           560   FORMAT(1X,'DAY OF THE YEAR=',3I1,1X,'TIME=',2I,I1,':',
114                 12Z1,':',2Z1,'.',3Z1)
115                 BOXES(1)=0
116                 BOXES(2)=0
117                 BOXES(3)=0
118                 BOXES(4)=0
119                 BOXES(5)=TOP
120           C     BOXES(33)=TOP
121           C     BOXES(5)=X
122                 IF(BOXES(5).LT.0)BOXES(5)=IAND(BOXES(5),'377'O)
123                 IF(BOXES(5).LT.1)BOXES(5)=1
124                 IF(BOXES(5).GT.254)BOXES(5)=254
125           C     BOXES(5)=BOXES(5).OR.'00140000'O
126                 BOXES(6)=BOTTOM
127           C     BOXES(34)=BOTTOM
128           C     BOXES(6)=BOXES(5)+10
129                 IF(BOXES(6).LT.0)BOXES(6)=IAND(BOXES(6),'377'O)
130                 IF(BOXES(6).GT.254)BOXES(6)=254
131                 BOXES(7)=LEFT
132           C     BOXES(35)=LEFT
133           C     BOXES(7)=Y
134                 IF(BOXES(7).LT.0)BOXES(7)=IAND(BOXES(7),'377'O)
135                 IF(BOXES(7).LT.1)BOXES(7)=1
136                 IF(BOXES(7).GT.242)BOXES(7)=242
137                 BOXES(8)=RIGHT
138           C     BOXES(36)=RIGHT
139                 IF(BOXES(8).LT.0)BOXES(8)=IAND(BOXES(8),'377'O)
140           C     BOXES(8)=BOXES(7)+10
141                 IF(BOXES(8).GT.242)BOXES(8)=242
142                 IF(TINDEX.EQ.0.AND.TFLAG.EQ.0)THEN
143                 ICHANGEF=0
144                 OBOX5=BOXES(5)
145                 OBOX7=BOXES(7)
146                 OBOX6=BOXES(6)
147                 OBOX8=BOXES(8)
148                 TFLAG=1
149                 ENDIF
150                 OBOX5P1=OBOX5+NOISE
151                 OBOX5M1=OBOX5-NOISE
152                 OBOX6P1=OBOX6+NOISE
153                 OBOX6M1=OBOX6-NOISE
154                 IF(OBOX5M1.GT.BOXES(5).OR.OBOX5P1.LT.BOXES(5)
155                 1.OR.OBOX6M1.GT.BOXES(6).OR.OBOX6P1.LT.BOXES(6))THEN
156           C     TYPE *,' TOP BOTTOM',OBOX5P1,OBOX5M1,OBOX7P1,
157           C     1OBOX7M1,BOXES(5),BOXES(7),TINDEX
158                 OBOX5=BOXES(5)
159                 OBOX6=BOXES(6)
160                 ICHANGEF=1
161                 ENDIF
162                 OBOX7M1=OBOX7-NOISE
163                 OBOX7P1=OBOX7+NOISE
164                 OBOX8M1=OBOX8-NOISE
165                 OBOX8P1=OBOX8+NOISE
```

```
166                     IF(OBOX7M1.GT.BOXES(7).OR.OBOX7P1.LT.BOXES(7)
167                     1.OR.OBOX8M1.GT.BOXES(8).OR.OBOX8P1.LT.BOXES(8))THEN
168             C       TYPE *,' LEFT RIGHT',OBOX5P1,OBOX5M1,
169             C       1OBOX7P1,OBOX7M1,BOXES(5),BOXES(7),TINDEX
170                     OBOX7=BOXES(7)
171                     OBOX8=BOXES(8)
172                     ICHANGEF=1
173                     ENDIF
174                     IF(ICHANGEF.EQ.1)THEN
175                     TINDEX=TINDEX+1
176             C       TYPE *,' TINDEX=',TINDEX
177                     IDAY=MO*100+TD*10+UD
178                     IH=TH*10+UH
179                     IM=TM*10+UM
180                     SEC=FLOAT(TS1*10+US)+FLOAT(TS2)/10.+FLOAT(LS)/100.
181                     1+FLOAT(MS)/1000.
182                     WRITE(9)(BOXES(IQ),IQ=5,8),IDAY,IH,IM,SEC
183                     ICHANGEF=0
184             C       WRITE(6,101)BOXES(5),BOXES(6),BOXES(7),BOXES(8)
185             101     FORMAT(1X,4(2X,O6))
186                     ENDIF
187             C       DO JK=9,40
188             C       BOXES(JK)=0
189             C       ENDDO
190                     K = SYSSQIO(%VAL(1),%VAL(GRCHAN),%VAL(%LOC(IO$_WRITEVBLK)),
191                     1IOSB,,,BOXES(1),%VAL(80),,,,)
192             100     CONTINUE
193                     GO TO 1
194                      END
195                     SUBROUTINE QASTT85
196                     BYTE C
197                     EXTERNAL IO$_READVBLK,IO$M_NOECHO
198                     EXTERNAL TT85AST
199                     INTEGER TTCHAN,SYSSQIO
200                     CHARACTER*15 STRING
201                     BYTE DATA(15)
202                     EQUIVALENCE(DATA,STRING)
203                     INTEGER*2 IOSB(4)
204                     COMMON/CHAN/TTCHAN
205                     K = SYSSQIO(%VAL(5),%VAL(TTCHAN),%VAL(%LOC(IO$_READVBLK)),IOSB,
206                     1TT85AST,C,DATA(1),%VAL(15),,,,)
207                     IF(.NOT.K)TYPE *,' ERROR IN QASTT85 QIO'
208                     RETURN
209                     END
210                     SUBROUTINE TT85AST(C)
211                     BYTE C
212                     EXTERNAL IO$_READVBLK,IO$M_NOECHO,IO$_WRITEVBLK
213                     INTEGER SYSSQIOW,ITCHAN,XRCHAN,IO$M_NOECHO,SYSSSETIMR
214                     INTEGER SYSSQIO,SYSSWAITFR,TSCHAN,SYSSBINTIM
215                     DOUBLE PRECISION QUAD
216                     CHARACTER*16 TIME
217                     DATA TIME/'0000 00:00:00.09'/
218             C       DATA TIME/'0000 00:00:01.09'/
219                     ISTATUS=SYSSBINTIM(%DESCR(TIME),QUAD)
220                     IF(.NOT.ISTATUS)TYPE *,' ERROR IN TIME DELAY'
```

```
221                 ISTATUS=SYS$SETIMR(%VAL(6),QUAD,,)
222                 IF(.NOT.ISTATUS)TYPE *,' ERROR IN TIME DELAY'
223                 ISTATUS=SYS$WAITFR(%VAL(6))
224                 IF(.NOT.ISTATUS)TYPE *,' ERROR IN TIME DELAY'
225                 CALL QASTT85
226                 RETURN
227                 END
228                 SUBROUTINE QXRA#(C)
229                 BYTE C
230           C     TYPE *,' XRA# AST SERVICE ROUTINE>>>>>>>>>>'
231                 RETURN
232                 END
```

APPENDIX D

PROGRAM TO EVALUATE GROUND TRUTH COMPLETENESS

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C       THIS PROGRAM TAKES THE GROUNDN.DAT FILE AND DISPLAYES THE BOXES
C       OUT ON THE HUMAN FACTORS HARDWARE SYNCHRONIZED WITH THE IRIG TRANSLATOR
C       AND/OR GENERATOR. FORMATTED FILES ARE NORMALLY CREATED WITH BOXPOS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        EXTERNAL IO$_READVBLK,IO$_WRITEVBLK
        PARAMETER NPOINTS=65525
        DOUBLE PRECISION QUAD
        CHARACTER*15 NAME,ISEARCH*1
        INTEGER SYS$WAITFR,SYS$BINTIM
        INTEGER*2 IRIG(6),IV(2),HD,TD,UD,TH,UH,TM,UM,TS1,TS2,US,LS,MS
        INTEGER*2 OBOX5,OBOX6,OBOX7,OBOX8,OBOX5M1,OBOX5P1,OBOX7M1
        INTEGER*2 OBOX7P1,IF1,TFLAG,TARGINDX(10)
        INTEGER*2 OBOX6M1,OBOX6P1,OBOX8M1,OBOX8P1,OPASS,BOXN
        BYTE TAPECNTRL
        INTEGER*2 IOSB(4),BOXES(40),BLINKM,ISAVE(2),IRIG2
        INTEGER*2 TOP,BOTTOM,LEFT,RIGHT
        INTEGER SYS$QIOW,SYS$ASSIGN,SYS$QIO,SYS$LKWSET
        INTEGER INLOCK(2),IOLOCK(2),XRCHAN
C       INTEGER*2 PASS,IHR,IMR,IH,IM
        BYTE IHR,IMR,IH,IM
        INTEGER*2 BOX(4,NPOINTS)
        BYTE TIH(NPOINTS),TIM(NPOINTS)
        REAL TSEC(NPOINTS)
        INTEGER*2 TIDAY(NPOINTS),IDAY
        EQUIVALENCE(BOXES(1),TOP),(BOXES(2),BOTTOM),(BOXES(3),LEFT)
        EQUIVALENCE(BOXES(4),RIGHT)
        DATA N/1/
        DATA IF1/0/
        DATA TINDEX/0/,TFLAG/0/
        ISTATUS=SYS$ASSIGN('XRA0',XRCHAN,,)
        IF(.NOT.ISTATUS)TYPE *,' ERROR IN XRA1 CHANNEL ASSIGN'
        TYPE *,' ENTER FORMATTED INPUT FILE NAME.(i.e. BOXPOS1.FTX)'
        READ(5,234)NAME
234     FORMAT(A)
        OPEN(UNIT=9,NAME=NAME,TYPE='OLD',READONLY)
        IC=0
        DO JK=1,40
        BOXES(JK)=0
        ENDDO
        K = SYS$QIOW(%VAL(1),%VAL(XRCHAN),%VAL(%LOC(IO$_WRITEVBLK)),
       1IOSB,,,BOXES(1),%VAL(80),,,,)
2       READ(9,221,END=100)IDAY,IHR,IMR,SECR,(BOXES(I),I=1,4)
221     FORMAT(1X,Z4,1X,I2,1X,I2,1X,F7.3,4(3X,I4))
C       BOXN=ISHFT(IDAY,-12)
C       WRITE(6,34)BOXN
C34     FORMAT(1X,Z4)
        IC=IC+1
        BOX(1,IC)=BOXES(1)
        BOX(2,IC)=BOXES(2)
        BOX(3,IC)=BOXES(3)
        BOX(4,IC)=BOXES(4)
        TIH(IC)=IHR
```

```
                TIM(IC)=IHR
                TSEC(IC)=SECR
                TIDAY(IC)=IDAY
C               WRITE(6,67)IH,IM,SEC
67              FORMAT(1X,I2,':',I2,':',F6.2)
                GO TO 2
100             TYPE *,'TOTAL NUMBER OF IRIGS=',IC
                L=1
                GO TO 1
765             TYPE *,' HIT RETURN TO CONTINUE IRIG SEARCH'
                READ(5,764)ISEARCH
764             FORMAT(A)
                L=1
1               K=SYS$QIOW(%VAL(1),%VAL(XRCHAN),%VAL(%LOC(IO$_READVBLK)),IOSB,
                1,,IRIG(1),%VAL(6),,,,)
C               WRITE(6,567)X,Y,IRIG(1),IRIG(2),IRIG(3)
567             FORMAT(1X,'X=',O3,5X,'Y=',O3,1X,O6,1X,O6,1X,O6)
                IF(IRIG(3))THEN
                L=1
                GO TO 1
                ENDIF
                CALL IRIGCVT(IRIG,IH,IM,SEC)
                LFLAG=0
C               TYPE *,' IH ,IM ,SEC =',IH,IM,SEC
2               IF(IH.LT.TIH(L))THEN
                        IF(L.GT.1)THEN
3                               L=L-1
                                LFLAG=1
                                IF(IH.LT.TIH(L).AND.L.GT.1)GO TO 3
                                GO TO 1
                        ENDIF
                        GO TO 1
                ENDIF
                IF(IH.EQ.TIH(L))GO TO 23
21              IF(LFLAG.EQ.1)GO TO 1
                L=L+1
                IF(L.GT.IC)THEN
                PRINT 223,TIH(L),TIM(L),TSEC(L)
223             FORMAT(1X,'IRIG NOT FOUND=',I2,':',I2,':',F6.2)
                PRINT 222,IH,IM,SEC
222             FORMAT(1X,'  CURRENT IRIG=',I2,':',I2,':',F6.2)
                IF(TIH(L).EQ.0.AND.TIM(L).EQ.0)GO TO 765
                L=1
                GO TO 1
                ENDIF
                IF(TIH(L).EQ.0)GO TO 21
                GO TO 2
23              IF(IM.LT. TIM(L))GO TO 1
C               IF(IM.NE.TIM(L))GO TO 21
                IF(IM.NE.TIM(L))THEN
                L=L+1
                GO TO 1
                ENDIF
                IF(SEC.LT.TSEC(L))GO TO 1
C               BOXN=ISHFT(TIDAY(L),-12)
```

```
111           C       WRITE(6,34)BOXN,L
112           C34     FORMAT(1X,Z4)
113                   BOXN=(ISHFT(TIDAY(L),-12)-1)*4
114           C       TYPE *,'BOXN=',BOXN,L,TIDAY(L)
115           C       PRINT 1234,BOXN,L,TIDAY(L)
116           1234    FORMAT(1X,'BOXN=',I3,2X,I6,2X,O6)
117                   BOXES(1+BOXN)=BOX(1,L)
118                   BOXES(2+BOXN)=BOX(2,L)
119                   BOXES(3+BOXN)=BOX(3,L)
120                   BOXES(4+BOXN)=BOX(4,L)
121                   K = SYS$QIO(%VAL(4),%VAL(XRCHAN),%VAL(%LOC(IO$_WRITEVBLK)),
122                   1IOSB,,,BOXES(1),%VAL(80),,,,)
123                   L=L+1
124                   GO TO 1
125                   END
126                   SUBROUTINE IRIGCVT(IRIG,IH,IM,SEC)
127                   BYTE IH,IM
128                   INTEGER*2 IRIG(6),HD,TD,UD,TH,UH,TM,UM,TS1,TS2,US,LS,MS
129                   INTEGER*2 ISAVE(2)
130           C       HD=ISHFT(IRIG(1),-12)
131           C       TD=IAND(ISHFT(IRIG(1),-8),'F'X)
132           C       UD=IAND(ISHFT(IRIG(1),-4),'F'X)
133                   TH=IAND(IRIG(1),'3'X)
134                   UH=ISHFT(IRIG(2),-12)
135                   TM=IAND(ISHFT(IRIG(2),-9),'7'O)
136                   UM=IAND(ISHFT(IRIG(2),-5),'F'X)
137                   TS1=IAND(ISHFT(IRIG(2),-2),'7'O)
138                   ISAVE(1)=IAND(ISHFT(IRIG(2),2),'C'X)
139                   ISAVE(2)=ISHFT(IRIG(3),-14)
140                   US=IOR(ISAVE(1),ISAVE(2))
141                   TS2=IAND(ISHFT(IRIG(3),-10),'F'X)
142                   LS=IAND(ISHFT(IRIG(3),-6),'F'X)
143                   MS=IAND(ISHFT(IRIG(3),-2),'F'X)
144           C       IRIG2=ISHFT(IRIG2,-2)
145                   IH=TH*10+UH
146                   IM=TM*10+UM
147                   SEC=FLOAT(TS1*10+US)+FLOAT(TS2)/10.+FLOAT(LS)/100.
148                   1+FLOAT(MS)/1000.
149                   RETURN
150                   END
```

APPENDIX E

DESCRIPTION OF TARGET DESIGNATOR/CUEING HARDWARE

DESCRIPTION OF TARGET DESIGNATOR/CUEING HARDWARE

The target designator/operator cueier or "Box Generator" has been designed to enable frame-by-frame target priority designation under computer control with video insertion built into the unit. The following features are available for use.

a. Ten targets can be cued at one time.

b. Each target can be outlined in white. The whole area inside (Box 8) will be increased in brightness.

c. There are five levels of prioritization cues available for use on the first 4 targets.

1. The highest level is when the blink bit is enabled on Box One. This will cause the box surrounding that target to go off and on.

2. There are four levels of brightness that can be set under computer control. This feature could, for example, be used to indicate a level of confidence in an automatic target selection process or to indicate the relative importance of different targets.

A block diagram of the overall acquisition system is shown in Figure E-1. The box representing the target designator/cueing hardware along with its different input-output signals is shown in the lower center of the figure. The following is a description of the use and meaning of data words into the target designator. The VAX-11/780 sends a 16 bit word through the DR11-B interface board to the target designator (TD). The first five high order bits are not used in the TD. Bit ten when high sets blink bit to Box 1. Box 1 is the only box that can be made to blink. Bit nine and bit eight control the brightness of four boxes, one through four. Bit value "0 0" is just brighter than the six boxes. Bit value "11" is the maximum brightness available. There can only be one box of each brightness. Bits seven through bit zero defines box position and box size. The size and position values are to be sent out in the following order, left side, right side, top side, and bottom side. The range and order of data is shown on the following page.

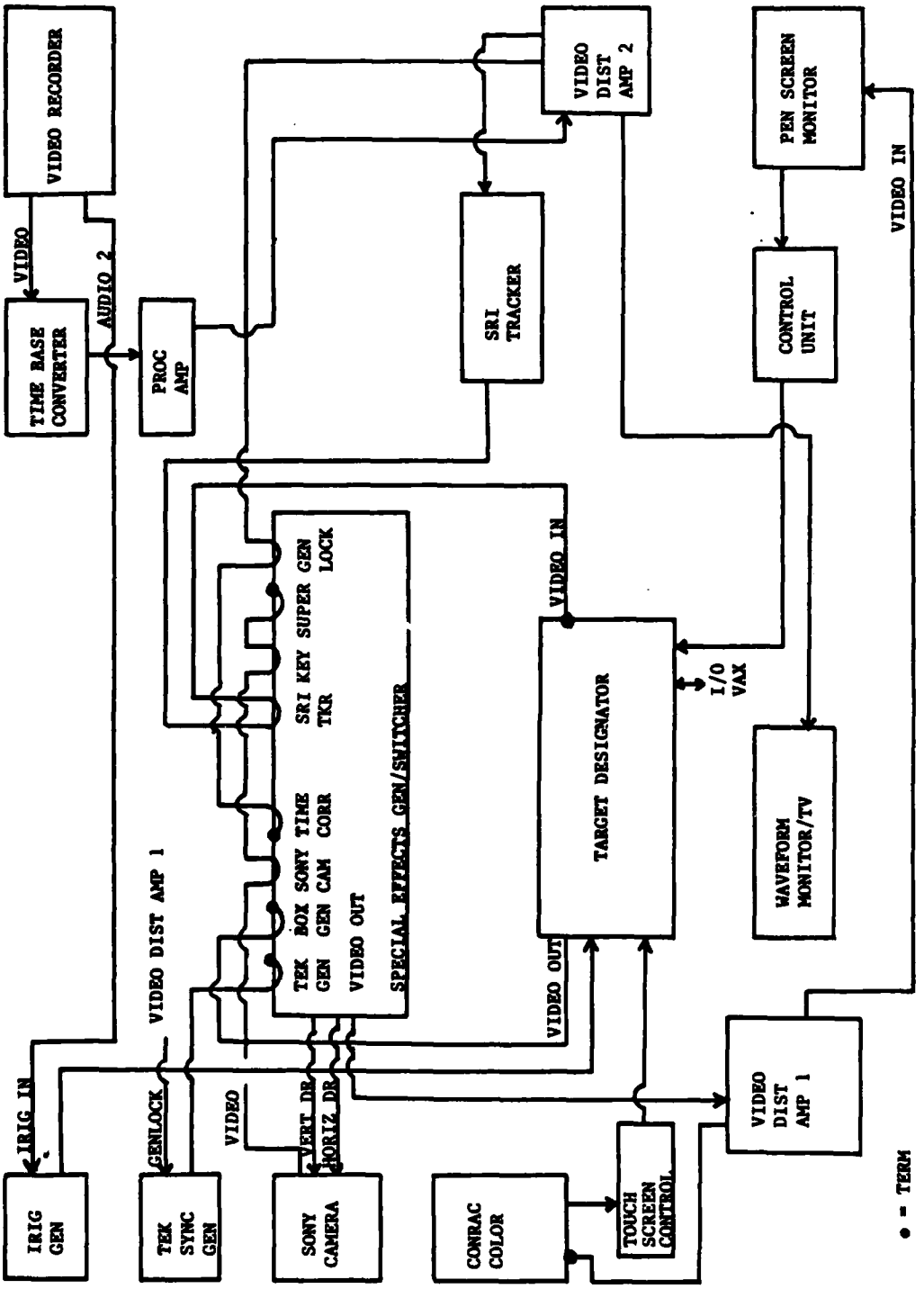
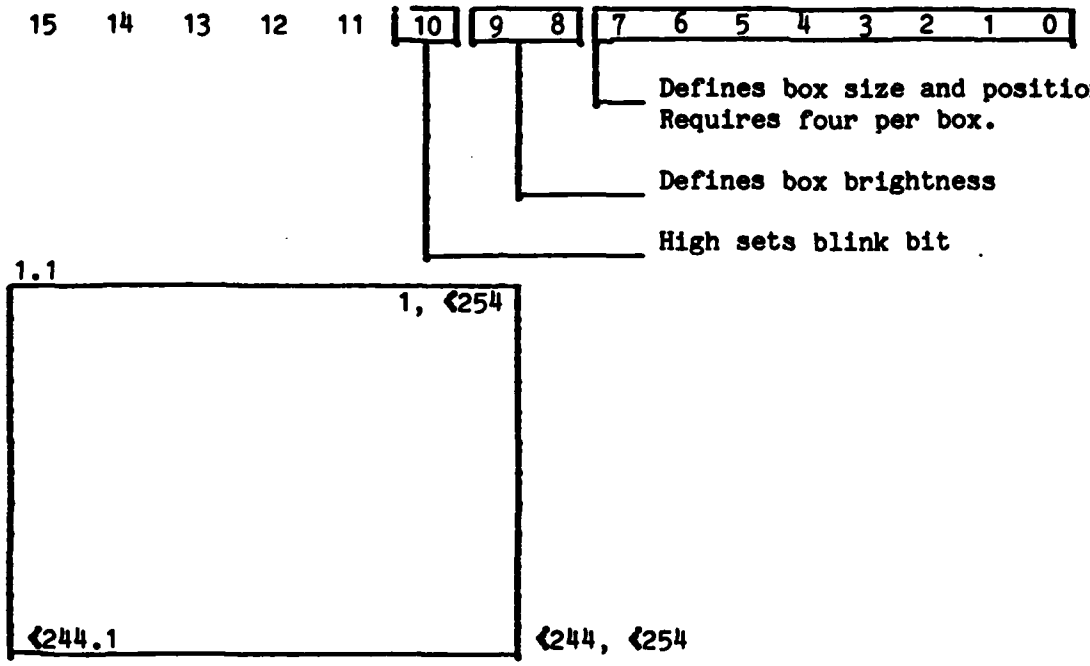


Figure E-1. System block diagram.



Values to set blink and brightness are set as part of left value for first four boxes. Blink is valid only for first box. Bits eight and nine have the following values.

Bit values for use for first four boxes, left side out only.

10	9	8	
0	0	0	No blink, minimum brightness
0	0	1	
0	1	0	
0	1	1	No blink, max brightness
1	0	0	Blink, box one only, minimum brightness
1	0	1	
1	1	0	
1	1	1	Blink, box one only, max brightness

RESET is initiated for 120 Nsec. from sync. interface control before up to forty unit values are output from DR11-B.

The following is a brief description of the target designator block diagram.

a. Latch address generation (Figure E-2)

A counter capable of storing more than forty numbers is first reset at beginning of sequence. This generates an address of zero. Data is put on the lines and LOAD DATA is initiated. This loads data into various latches that represent left side, blink and brightness. INC. then increments the latch address counter. This is continued until forty data words are loaded.

b. Data Reg and comparators (Figure E-3)

The data registers are loaded under the control of the latch address counter. The data register outputs are connected to high speed comparators where the values are compared against pixel count and line count. These comparators give signals out that indicate equal-to, greater-than, or less-than, for left, right, up and down. The signals are combined in high speed logic to derive boxes that are inserted into the video to indicate target position, size, and importance.

c. Pixel and line counters (Figure E-4)

The pixel and line counters generate counts corresponding to position on the TV screen. The composite blanking and vertical drive signals must be phase locked to the video signal of interest. These counter values are then sent to the comparators to generate box size and position.

d. Brightness select, blink, and video insert (Figure E-5)

Four unique select signals from left edge load on the first four boxes causes data bits on eight, nine, and ten to cause blinking on one and intensity modulation on one through four. These signals are then fed into a video amplifier where they are inserted into the video.

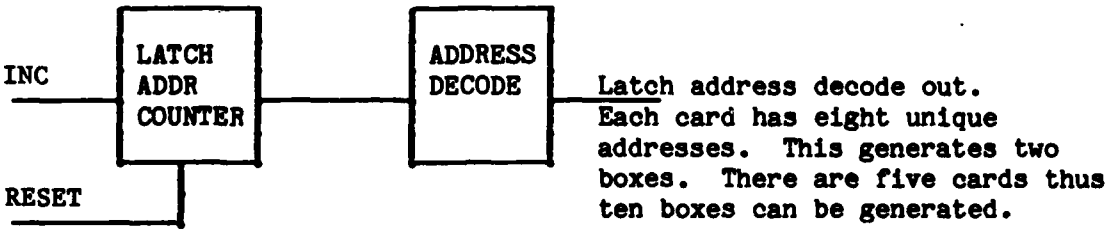


Figure E-2. Latch address generation.

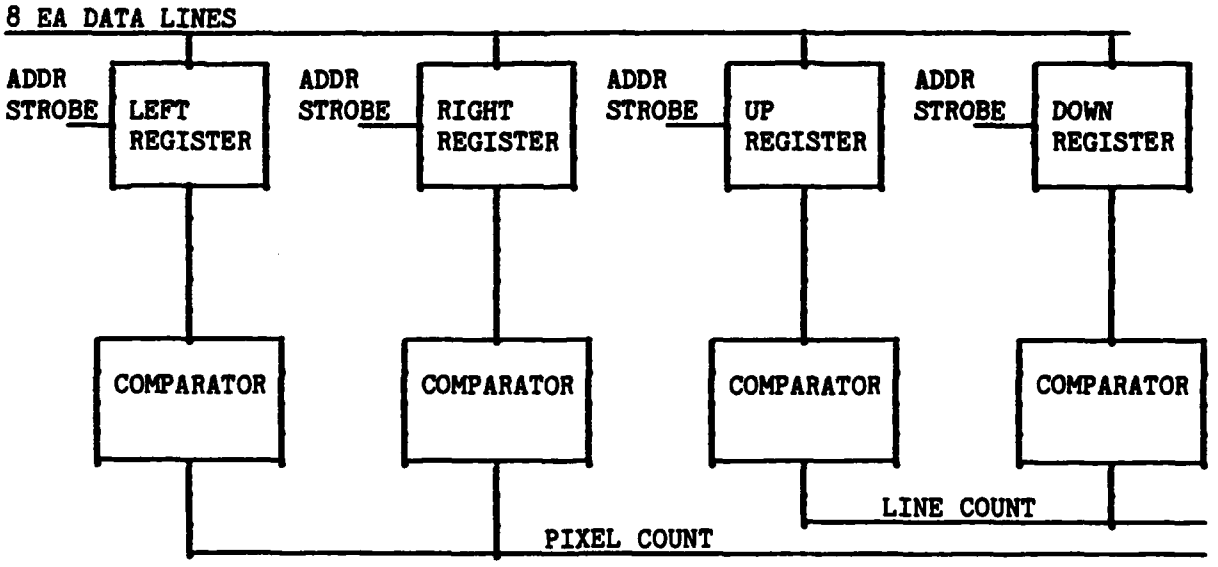


Figure E-3. Data reg and comparators.

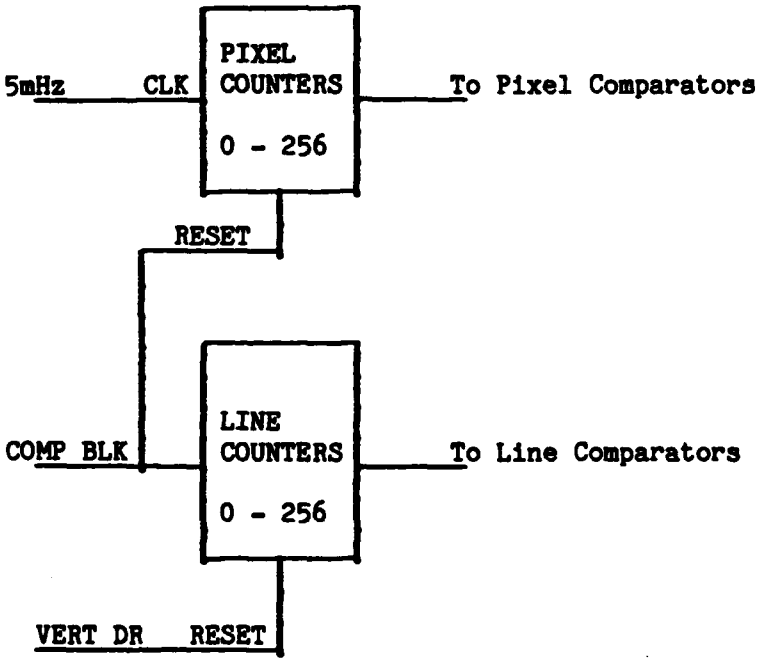


Figure E-4. Pixel and line counters.

Four unique selects from left
edge load on first four boxes

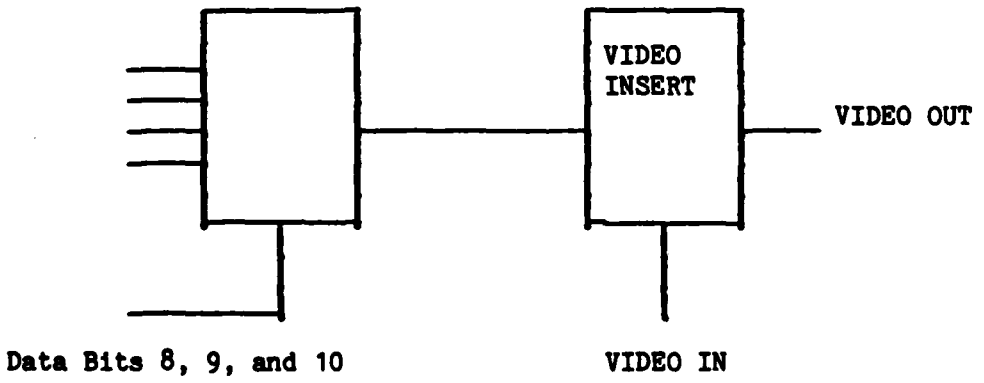


Figure E-5. Brightness select, blink, and video insert.

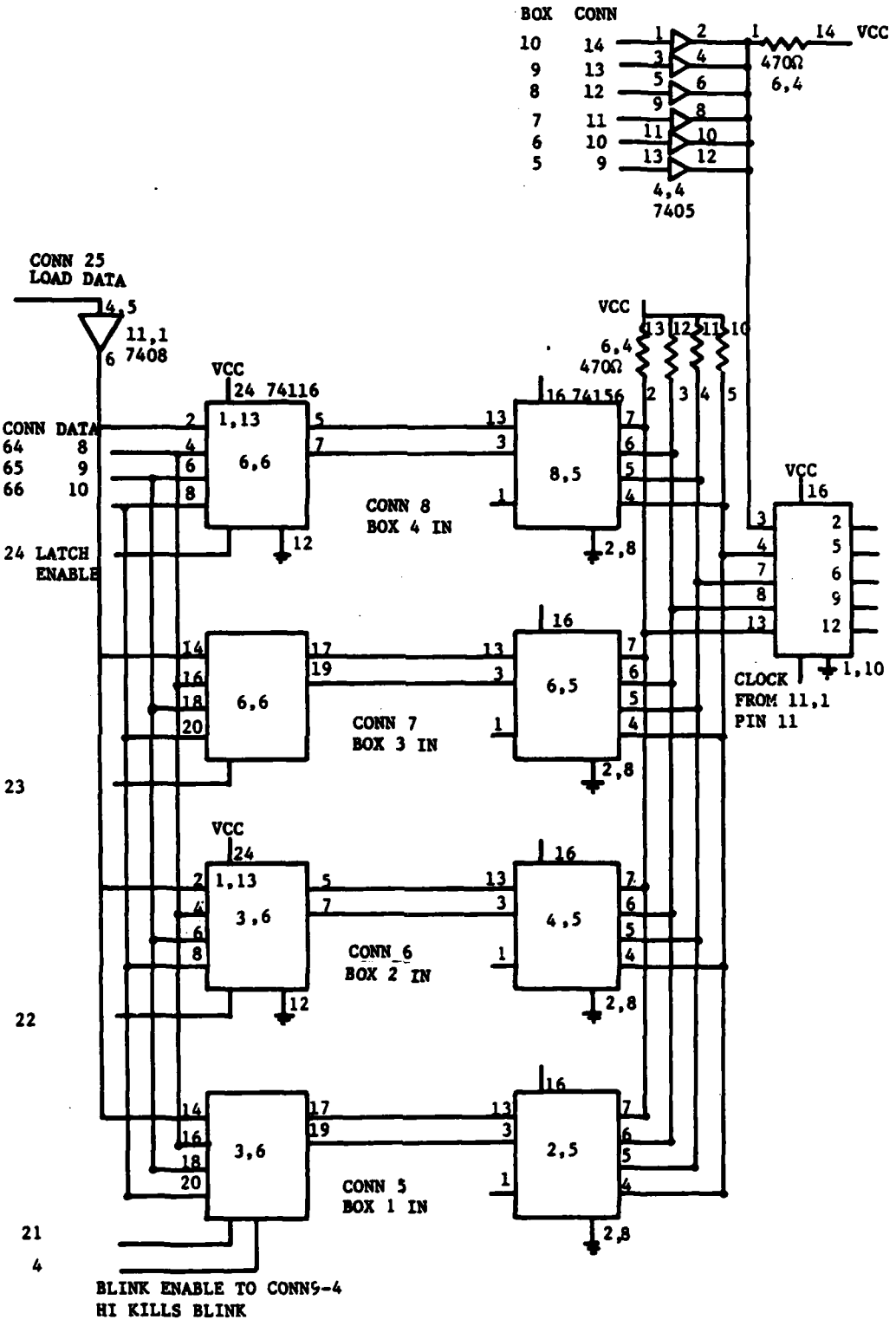
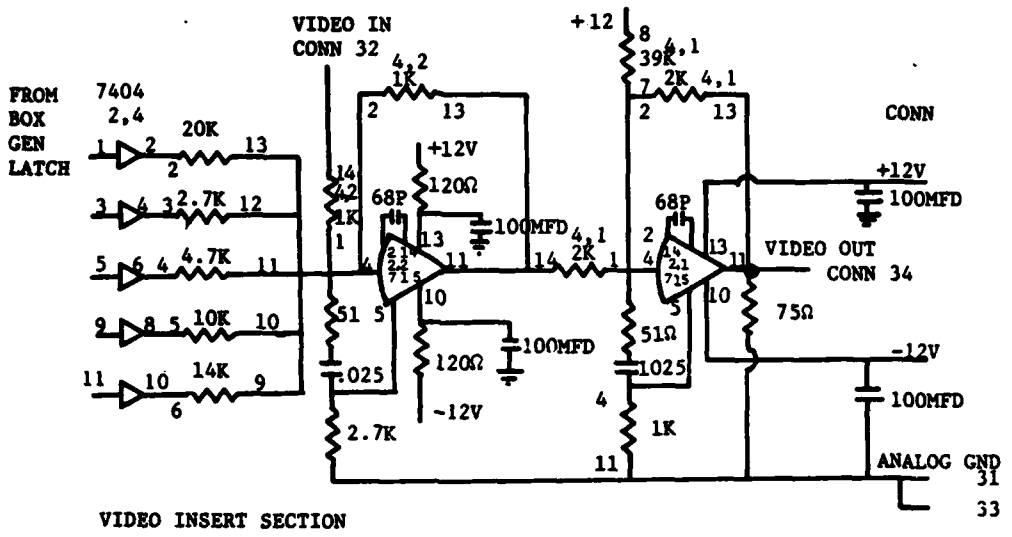


Figure E-6. Schematic of video insert, line and pixel.



VIDEO INSERT SECTION

Figure E-6. Continued.

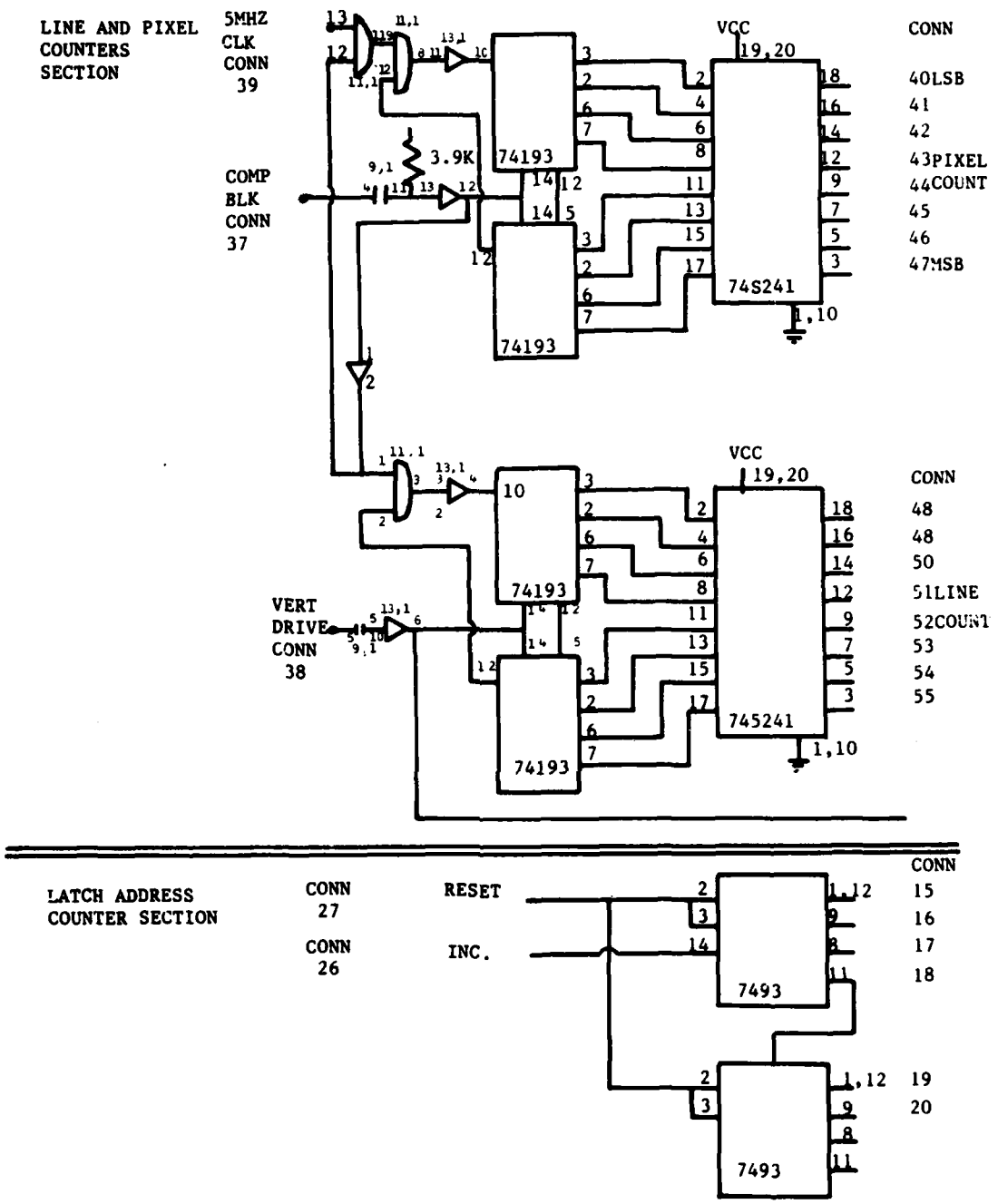
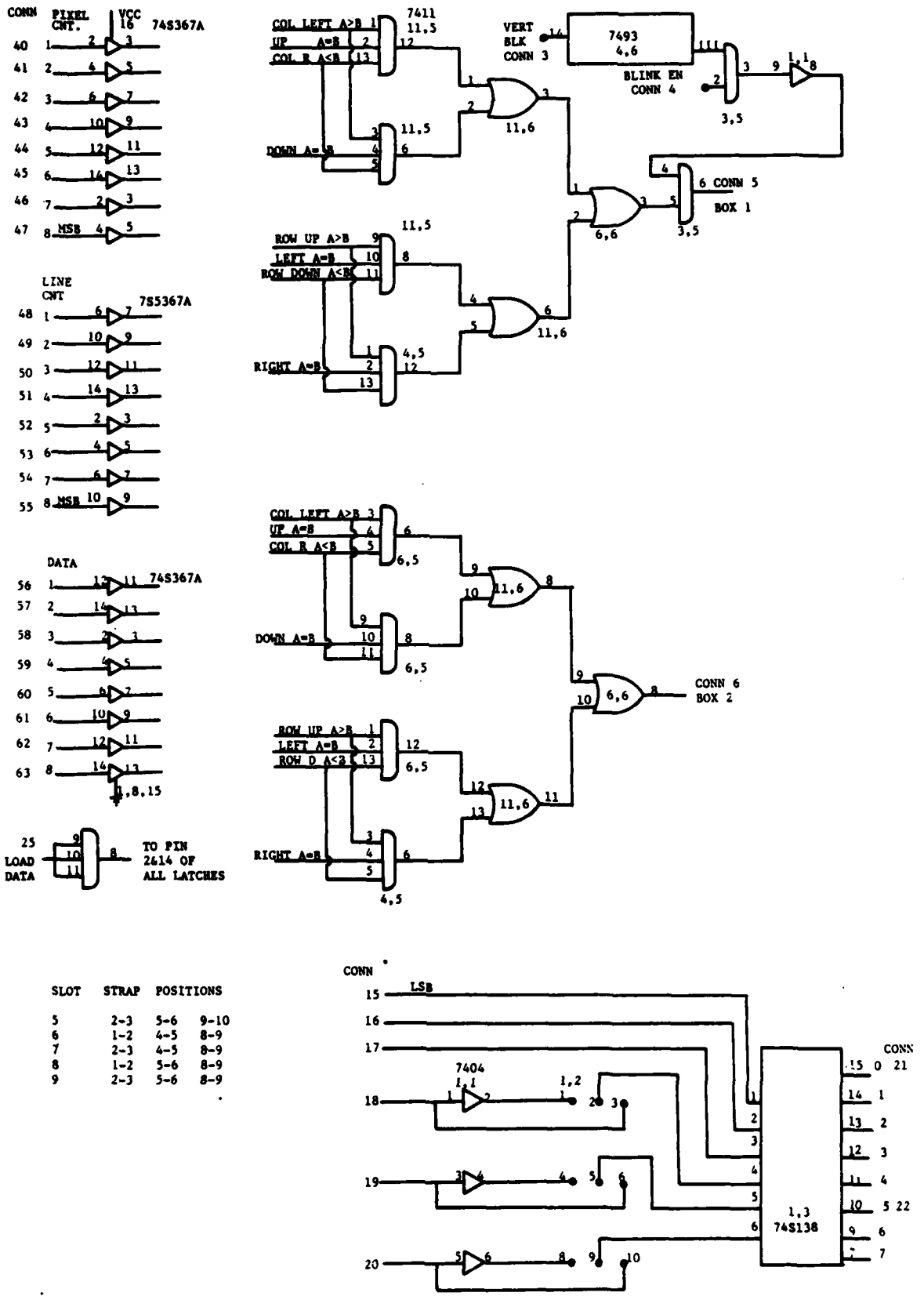
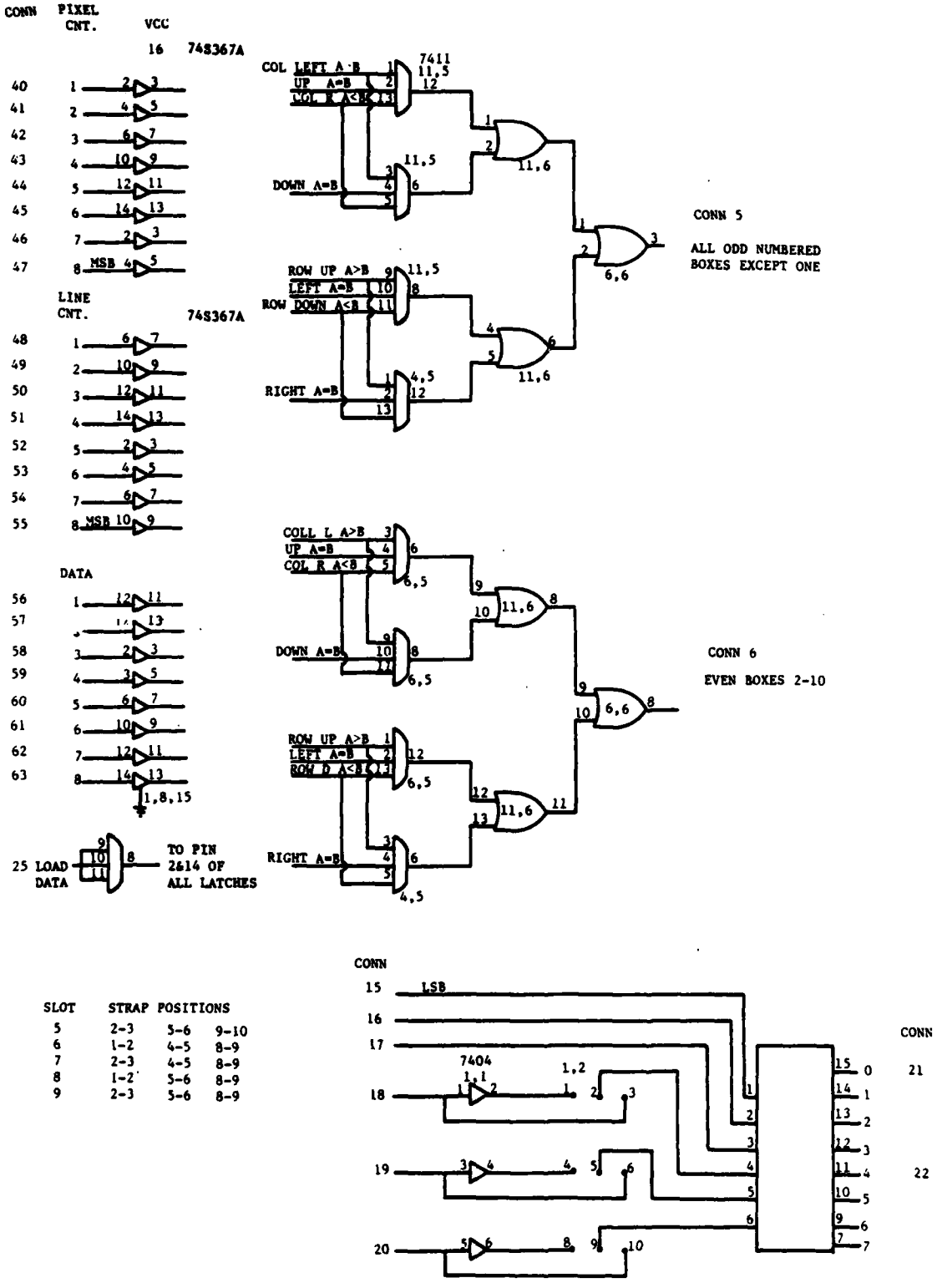


Figure E-6. Continued.



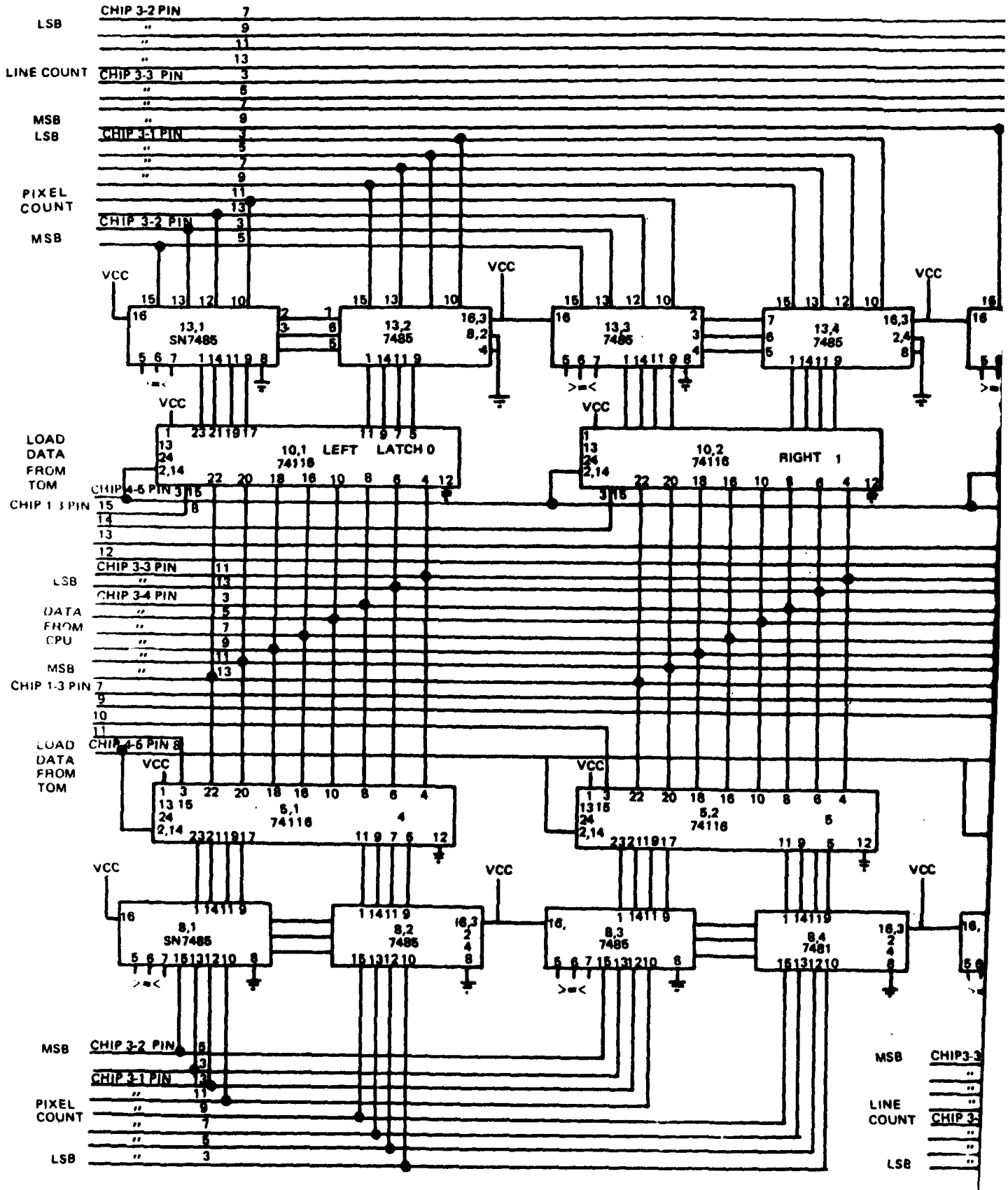
SLOT	STRAP	POSITIONS	
5	2-3	5-6	9-10
6	1-2	4-5	8-9
7	2-3	4-5	8-9
8	1-2	5-6	8-9
9	2-3	5-6	8-9

Figure E-7. Schematic of address select and box gen logic.



SLOT	STRAP POSITIONS		
5	2-3	5-6	9-10
6	1-2	4-5	8-9
7	2-3	4-5	8-9
8	1-2	5-6	8-9
9	2-3	5-6	8-9

Figure E-7. Continued.


LSB
LINE COUNT
MSB
LSB
PIXEL COUNT
MSB
CHIP 3-2 PIN
CHIP 3-3 PIN
CHIP 3-1 PIN
VCC
13,1 SN7485
13,2 7485
13,3 7485
13,4 7485
10,1 74116 LEFT LATCH 0
10,2 74116 RIGHT 1
LOAD DATA FROM TOM
CHIP 4-6 PIN 3
CHIP 1-3 PIN
DATA FROM CPU
CHIP 3-4 PIN
CHIP 4-6 PIN 8
5,1 74116 4
5,2 74116 5
8,1 SN7485
8,2 7485
8,3 7485
8,4 7481
CHIP3-3
CHIP 3-

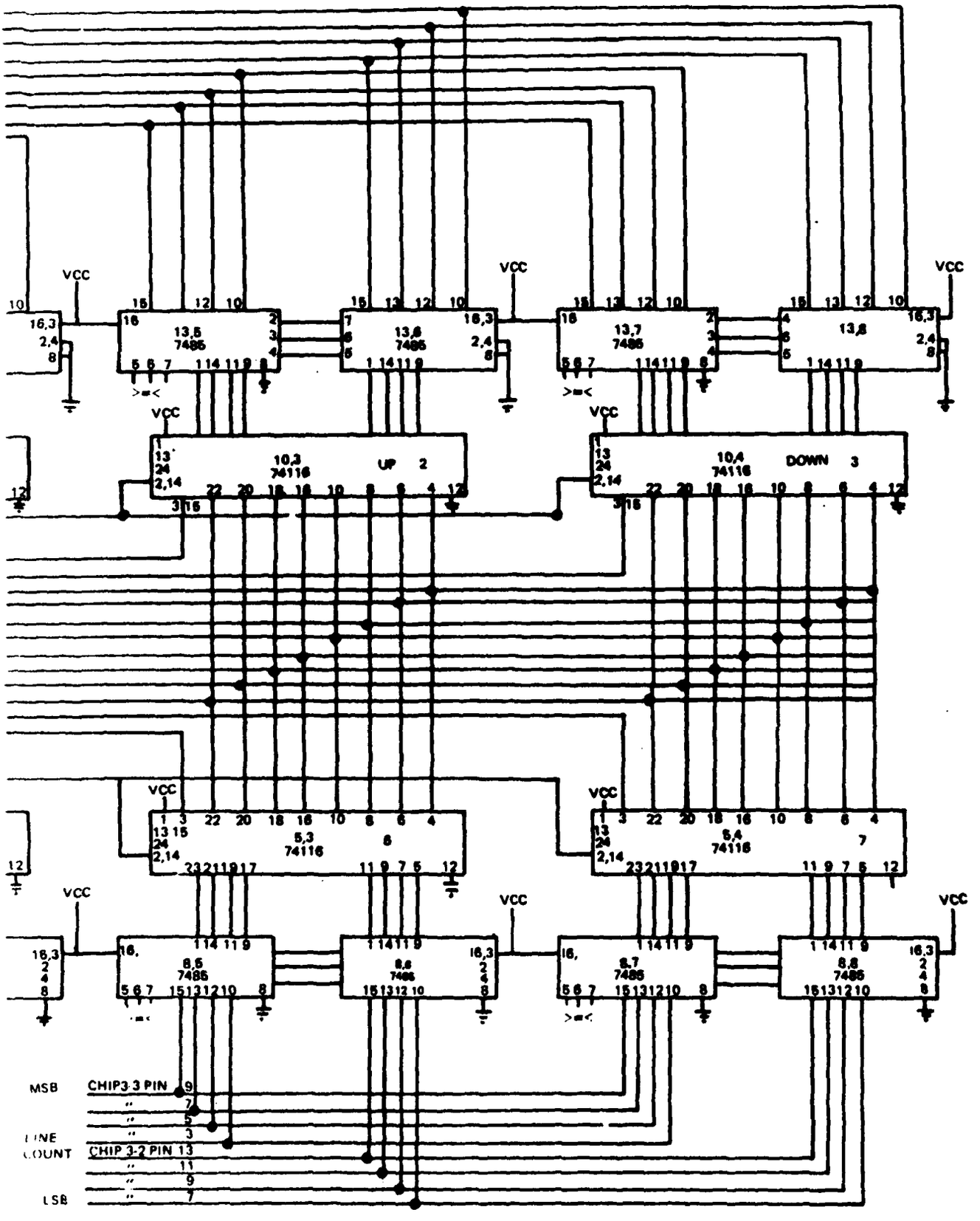


Figure E-8. Schematic data registers and comparators.

292

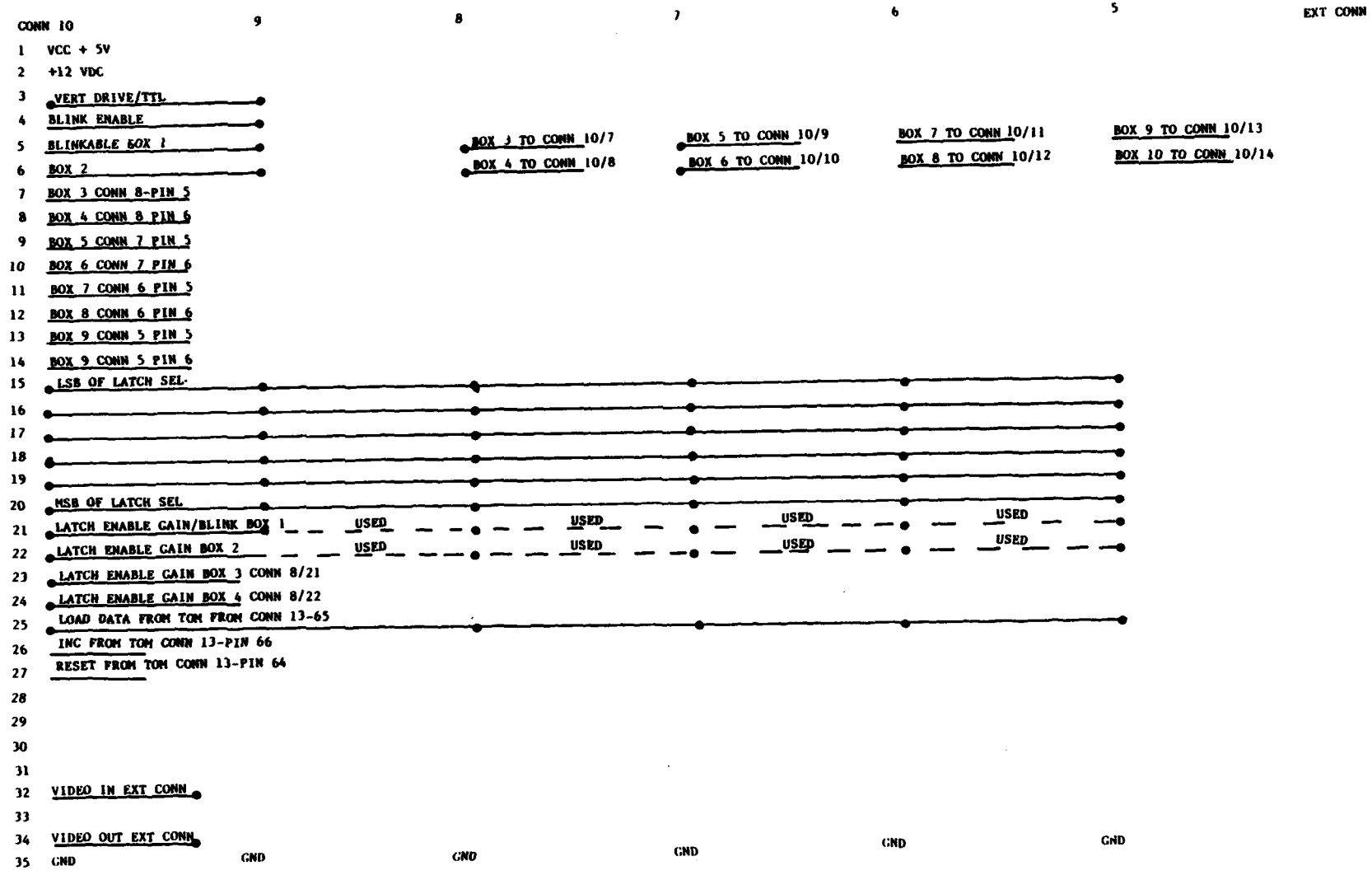


Figure E-9. Wiring of back plane.

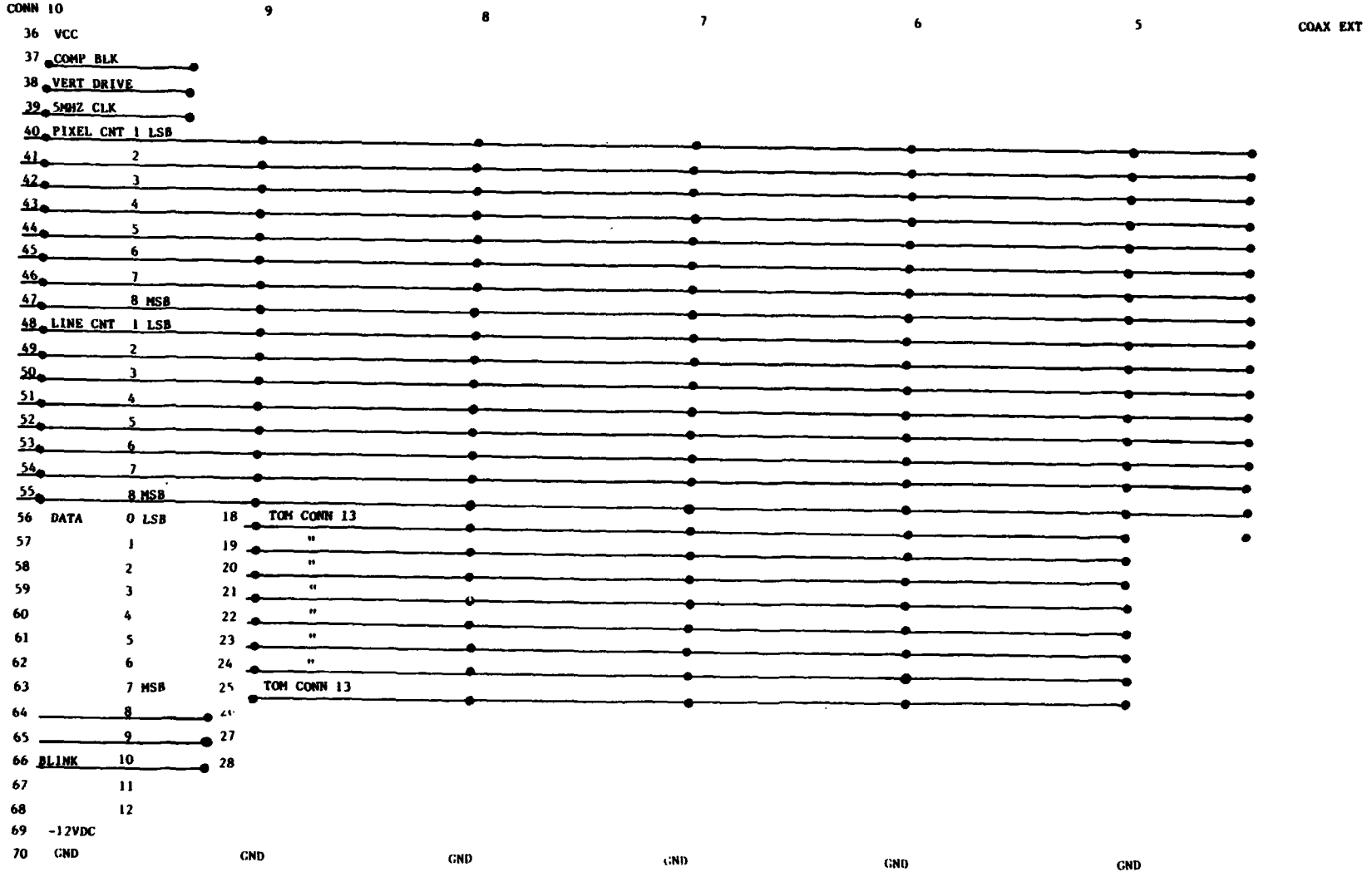


Figure E-9. Continued.

APPENDIX F

DESCRIPTION OF TRACKER INTERFACE AND CONTROL UNIT

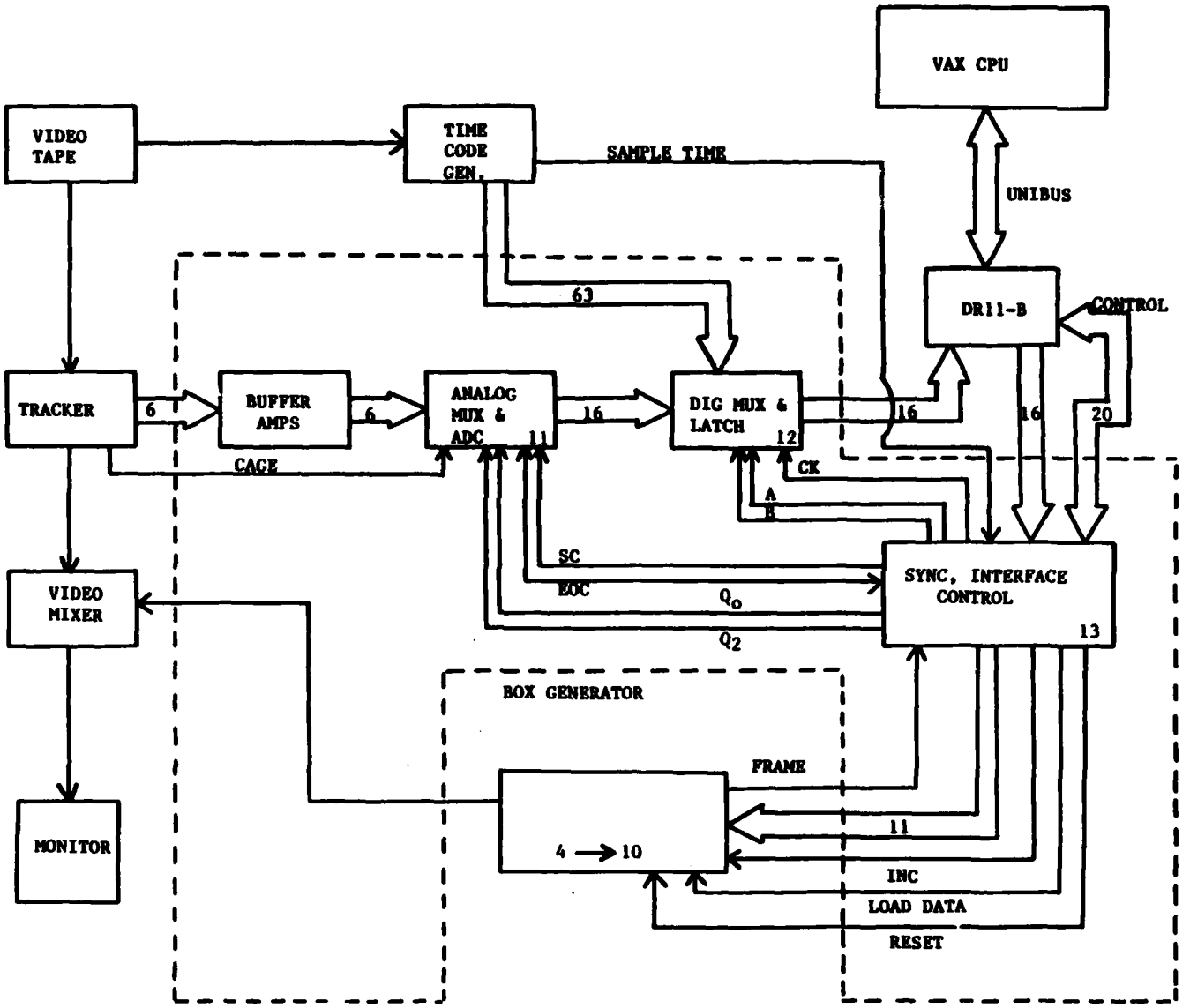


Figure F-1. Interface and control unit block diagram.

TRACKER INTERFACE AND CONTROL UNIT

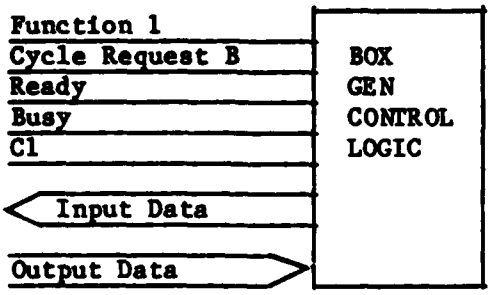
A block diagram of the interface and control unit along with its relation to other elements in the data acquisition system is shown in Figure F-1. This interface is fabricated on three separate circuit cards — a sync interface control board, a digital converter/multiplexer and latch board, and an analog to digital converter/multiplexer. The key element is the SYNC interface card. The SYNC-interface card performs the basic functions listed below.

- Generation of SYNC and timing signals for the Box-GEN.
- Interfacing of data and instructions output from the computer to the box-GEN.
- Interfacing of data input to the computer from the box-GEN.

The following signals are required for the transfer of data between the DR11-B and the box-GEN.

Signal	LOGIC 0 (Inactive)	LOGIC 1 (Active)
C1 (read/write, word)	0 VDC	+5 VDC (TTL)
Cycle Request B	0 VDC	+5 VDC (TTL)
Function 1 (read/write)	0 VDC	+5 VDC (TTL)
Ready	+5 VDC	0 VDC (TTL)
Busy	0 VDC	+5 VDC (TTL)
Output Data (16 lines)	0 VDC	+5 VDC (TTL)
Input Data (16 lines)	0 VDC	+5 VDC (TTL)

Box generator unidirectional interface to a DEC DR11-B



Function 1: This signal is returned as C1 control through two inverters and cable driver. When active, it indicates that the data transfer will be an input to the CPU from the Box GEN. When inactive, it indicates that the data transfer will be an output from the CPU to the Box-GEN.

Ready: When this signal becomes false (inactive), the DR11-B is ready to transfer data to/from the Box-GEN. When Function 1 is inactive, the Box-GEN will respond with a cycle request B to indicate it can accept an output data transfer. When function 1 is active, the Box-GEN will respond with a cycle request B to indicate it has an input data word ready for transfer.

Busy: This signal indicates that a bus sequence is in progress. The trailing edge of this signal is used to initiate subsequent output/input data requests. This trailing edge is also used to load output data into the Box-GEN during output data transfers.

Output Data Bus: The 16-bit positive true data bus output from the computer is input to card 13 at right side of page. Rising edge of "LOAD DATA" and "INC" on card 13 is used to load data in registers and increment address. "RESET" resets address register when ready first goes low for 120 nsec.

Input Data Bus: The 16 bit positive true data bus input to the computer is on card 12 left side of page. Data is latched and multiplexed on this card. ADC inputs come from intercard cable. The falling edge of busy increments two counters on card 13 for address generation. Address "A" and "B" are used for time code digital multiplexing and address "Qo" and "Q2" are used for analog multiplexing on input of ADC Card 11. (Refer to timing diagram in read mode.) The first start convert (SC) is generated during address 0 and cycle request. The last two start converts begin at falling edge of busy and the last two cycle request begin at end of convert (EOC). A BCD counter is used to select number of frame delays before generating first cycle request. The input to the ADC multiplexer has 6 buffer amps. Each buffer amp has balance and amplitude POT adjustments.

Figures F-2 through F-11 are the schematic and timing diagrams for the cards in the interface and control unit.

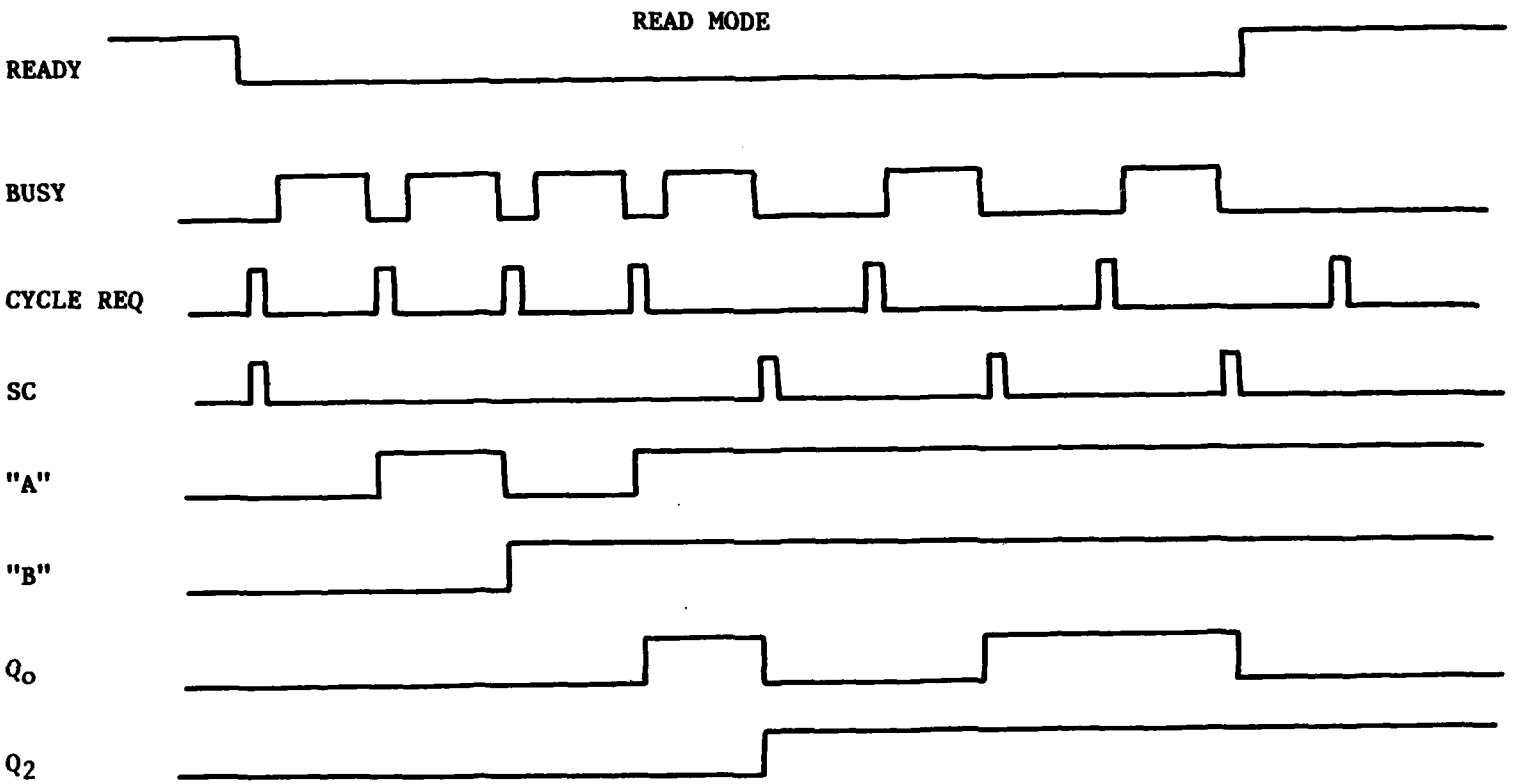
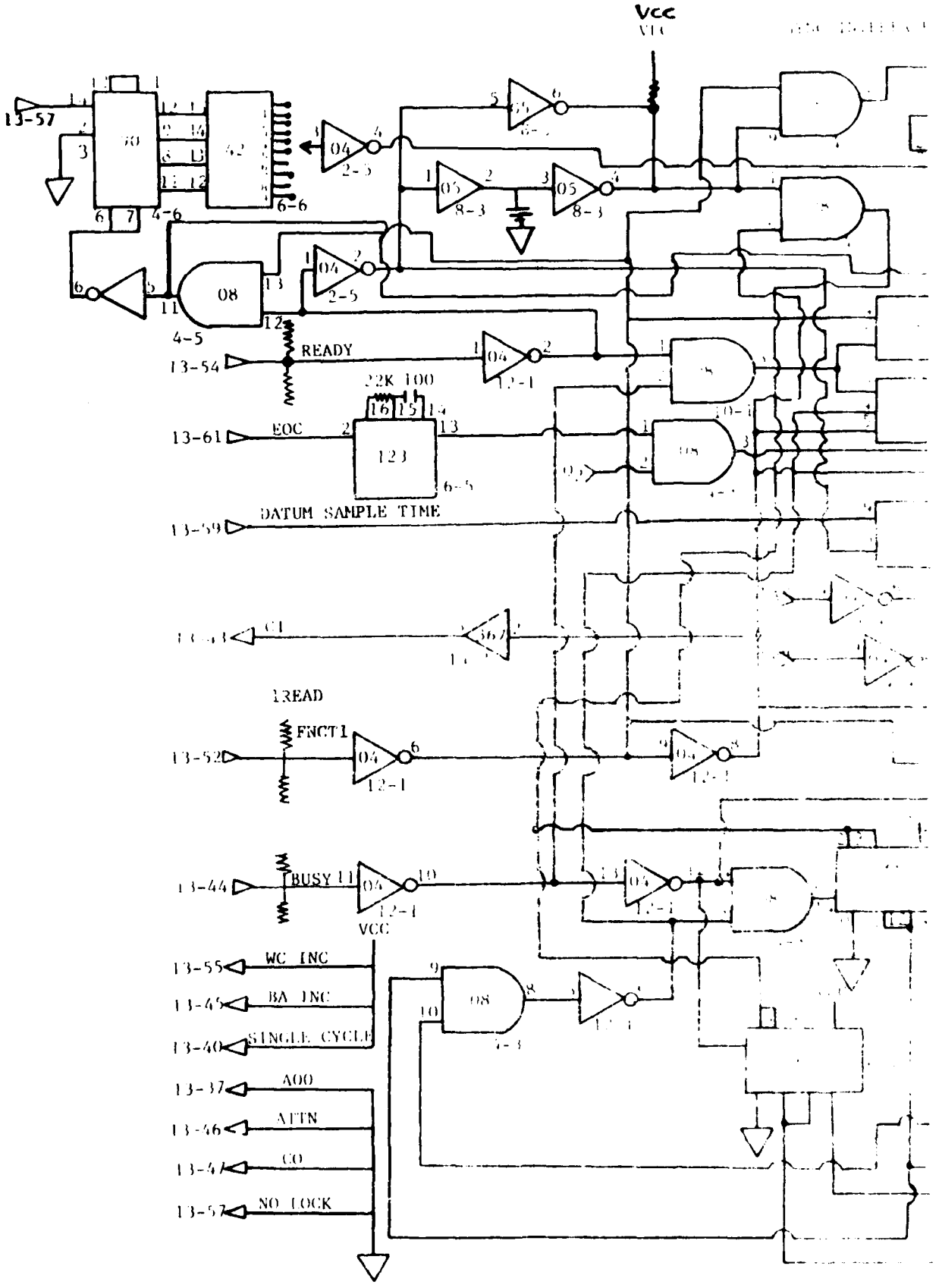


Figure F-2. Read mode timing diagram.


Vcc
13-57
READY
13-54
22K 100
13-61
EOC
123
DATUM SAMPLE TIME
13-59
13-53
CI
1READ
FNCT1
13-52
13-44
BUSY L1
VCC
13-55
WC INC
13-45
BA INC
13-40
SINGLE CYCLE
13-37
A00
13-46
ATTN
13-47
CO
13-57
NO LOCK

182

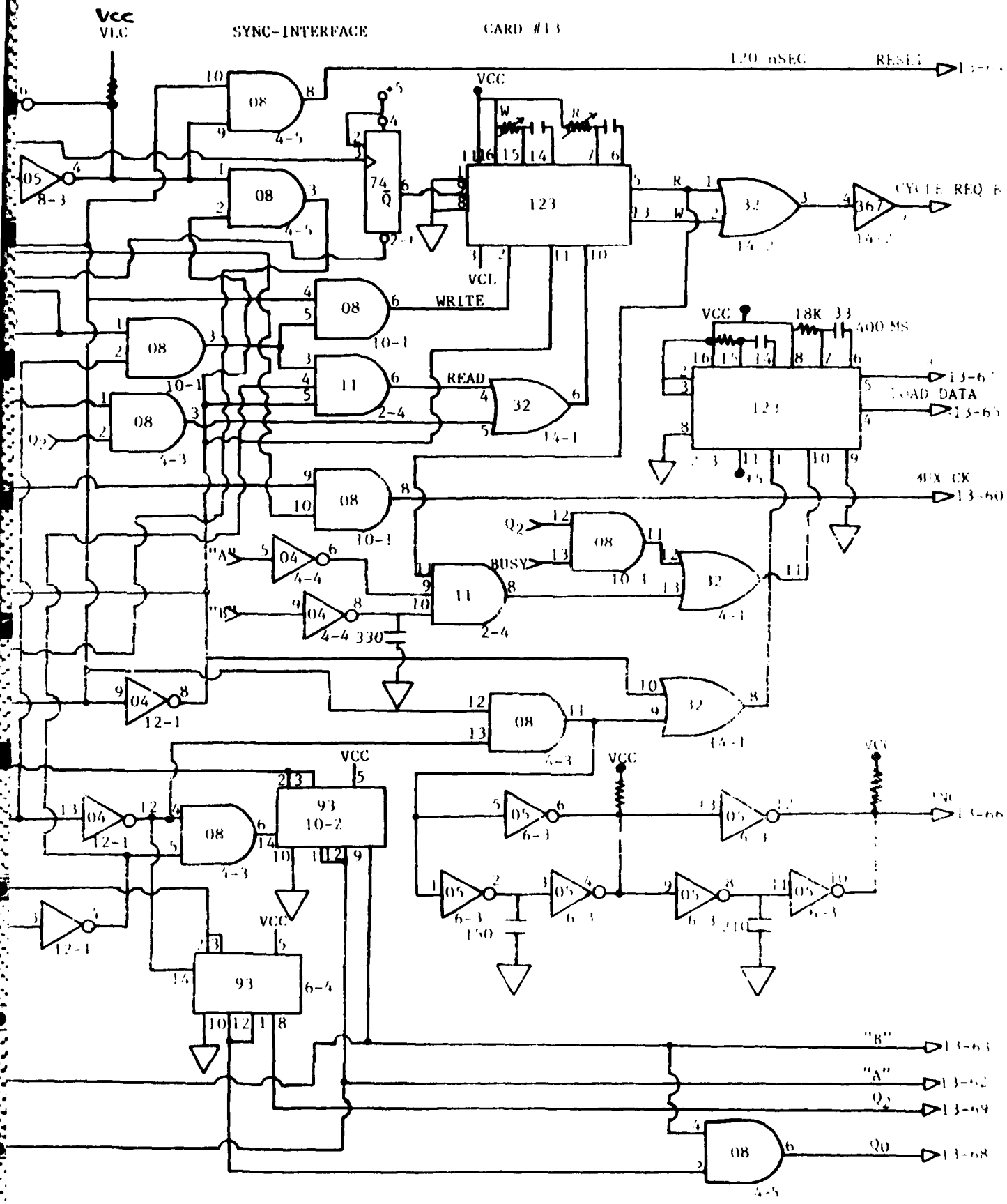


Figure F-3. Sync-Interface, card #13 schematic sheet 1 and sheet 2.

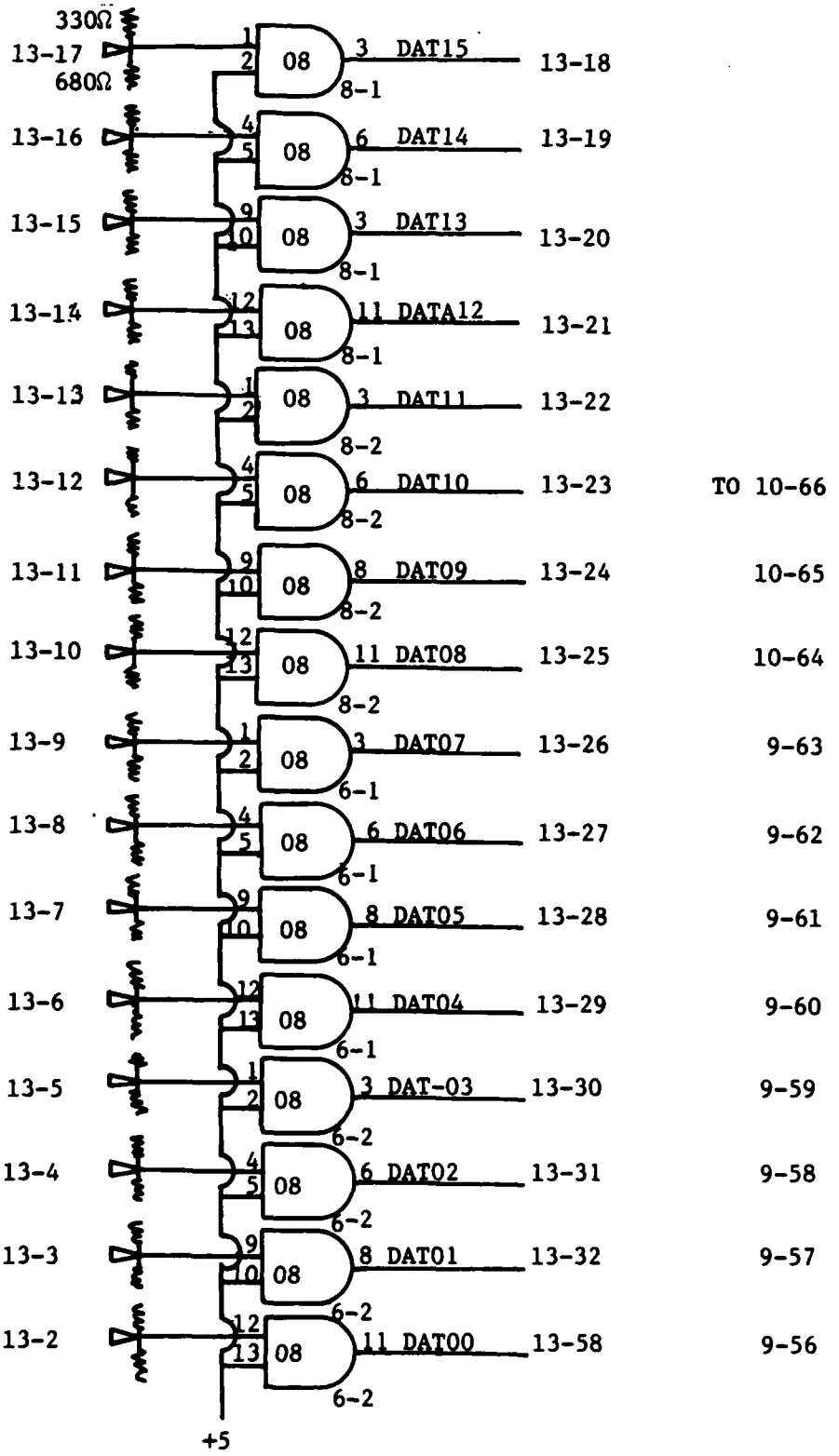


Figure F-3. Continued.

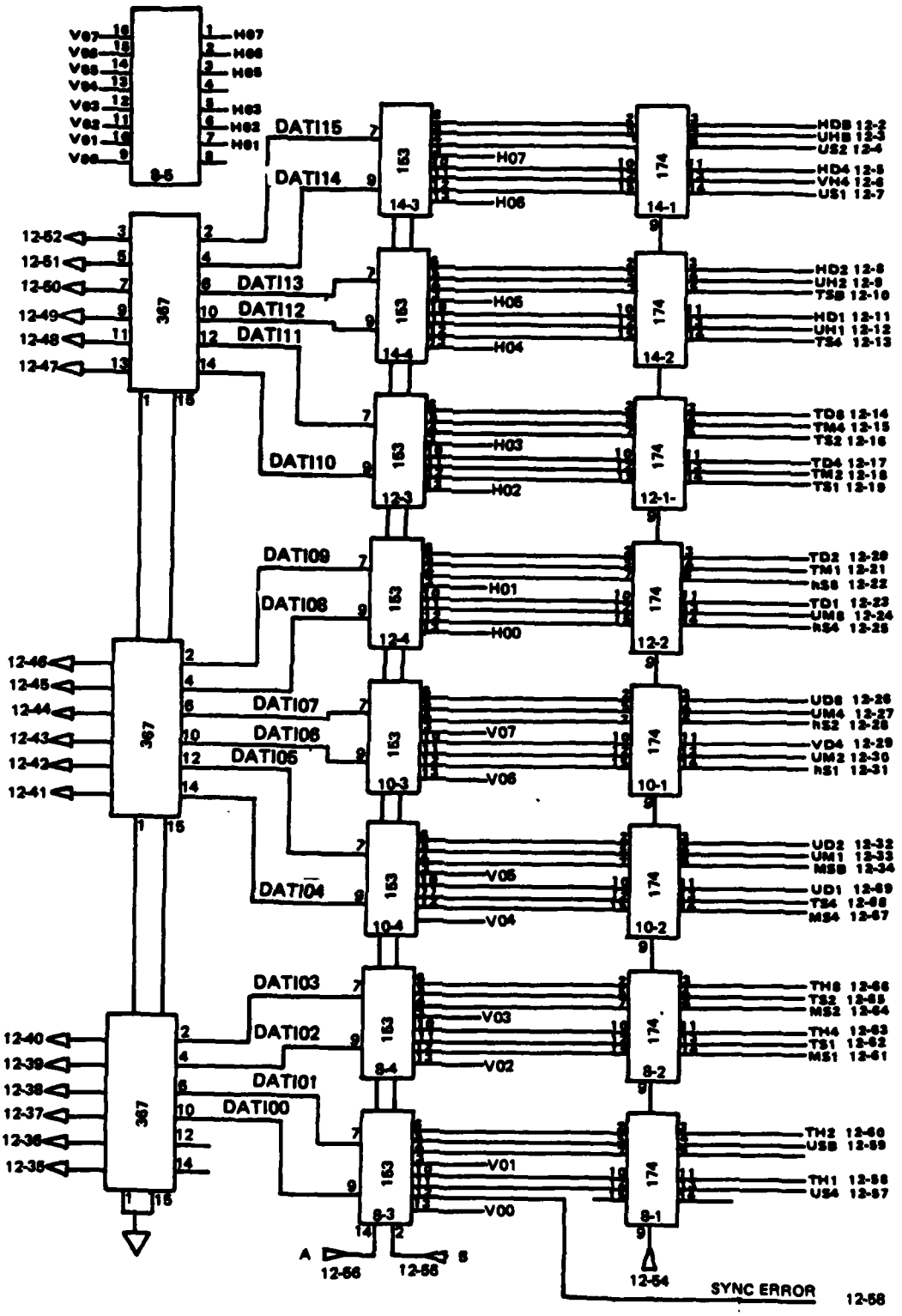


Figure F-4. Latch-multiplexer-cable driver, card #12 schematic.

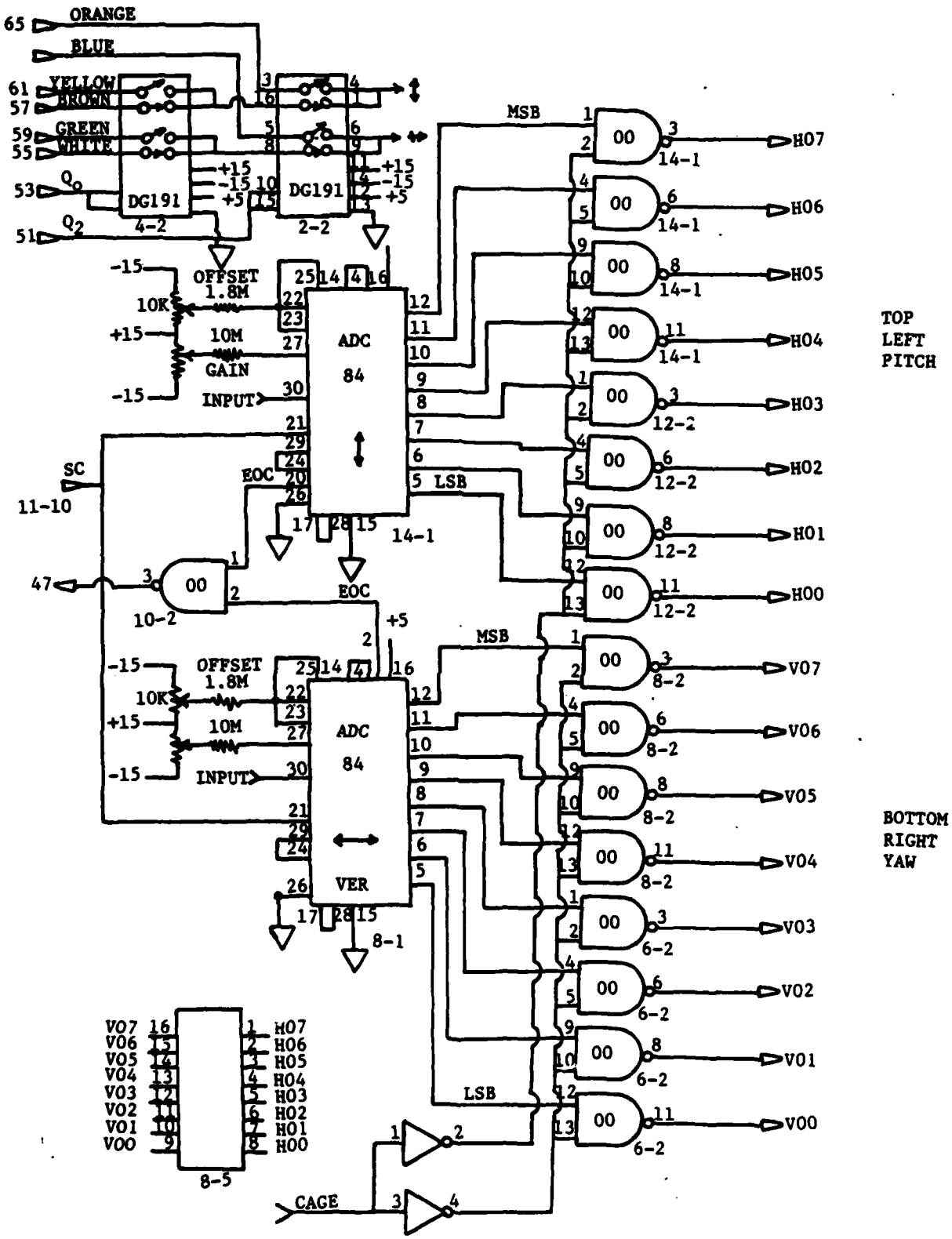


Figure F-5. Analog-multiplexer ADC, card #11 schematic.

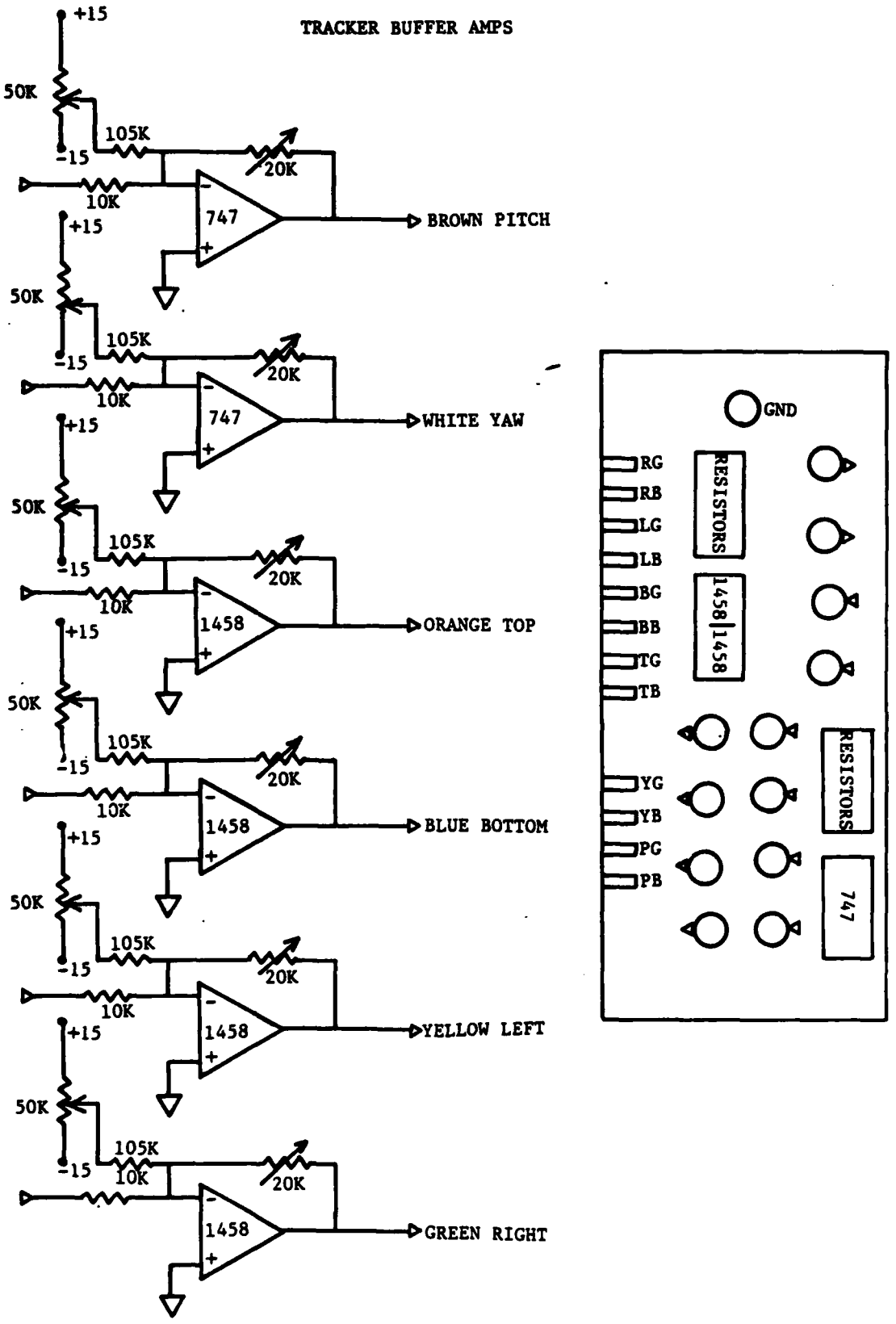


Figure F-6. Tracker buffer amplifier schematic.

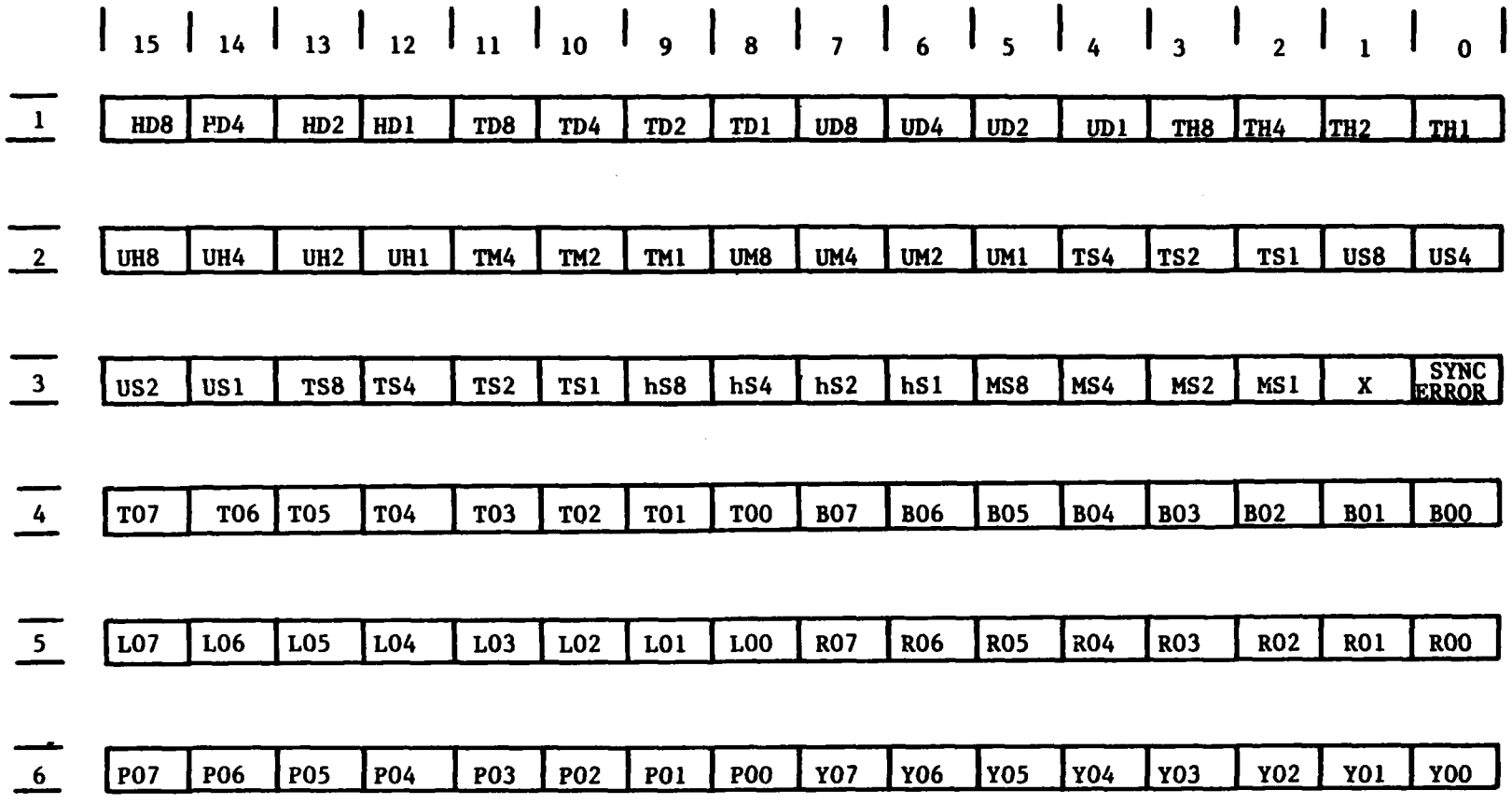
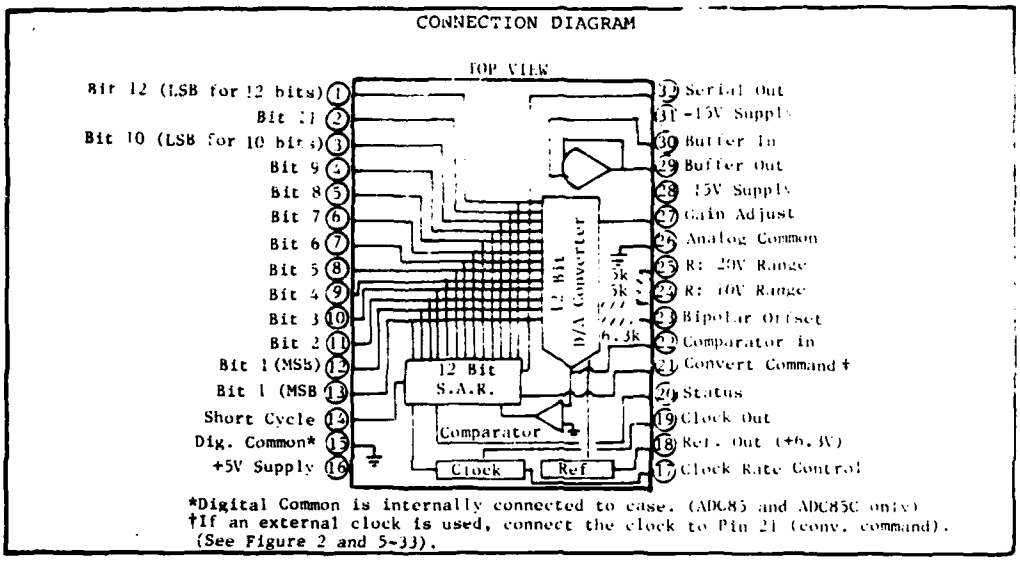


Figure F-7. Read data format diagram.



*Digital Common is internally connected to case. (ADC85 and ADC85C only)
†If an external clock is used, connect the clock to Pin 21 (conv. command). (See Figure 2 and 5-33).

Binary (BIN) Output	INPUT VOLTAGE RANGE AND LSB VALUES					
Analog Input Voltage Range	Defined As:	±10V	+5V	±2.5V*	0 to +10V	0 to +5V
Code Designation		COB* or CTC***	COB* or CTC***	COB* or CTC***	CSB**	CSB**
One Least Significant Bit (LSB)	FSR 2* n=8 n=10 n=12	20V 2* 78.13mV 19.53mV 4.88mV	10V 2* 39.06mV 9.77mV 2.44mV	5V 2* 19.53mV 4.88mV 1.22mV	10V 2* 39.06mV 9.77mV 2.44mV	5V 2* 19.53mV 4.88mV 1.22mV
Transition Values MSB LSB 000 . . . 000**** 011 . . . 111 111 . . . 110	+Full Scale Mil Scale -Full Scale	+10V-3 2LSB 0 +10V+1 2LSB	+5V-3 2LSB 0 -5V+2LSB	+2.5V-3 2LSB 0 -2.5V+1 2LSB	+10V-3 2LSB +5V 0+1 2LSB	+5V-3 2LSB +2.5V 0+1 2LSB

*COB=Complementary Offset Binary
**CSB=Complementary Straight Binary
***CTC=Complementary. Two's complement-obtained by using the complement of the most significant bit (MSB is available on pin 13.
**** Voltages given are the nominal value for transition to the code specified.

*USED IN THIS DESIGN

Input Signal Range	Output Code	Connect Pin 23 To Pin	Connect Pin 25 To	For Buffered Input* Connect Pin 29 To Pin	For Direct Input (See note) Connect Input Signal To Pin
+10V	COB or CTC	22	Input Signal**	25	25
+5V	COB or CTC	22	Open	24	24
±2.5V	COB or CTC	22	Pin 22	24	24
0 to +5V	CSB	26	Pin 22	24	24
0 to +10V	CSB	26	Open	24	24

*Connect to Pin 29 or input signal as shown in next two columns.
**The input signal is connected to Pin 30 if the buffer amplifier is used.
NOTE: If the buffer amplifier is not used, the input Pin 30 must be grounded (Pin 26).

TABLE I. INPUT VOLTAGES, TRANSITION VALUES, LSB VALUES, AND CODE DEFINITIONS

Figure F-8. A-D converter specifications.

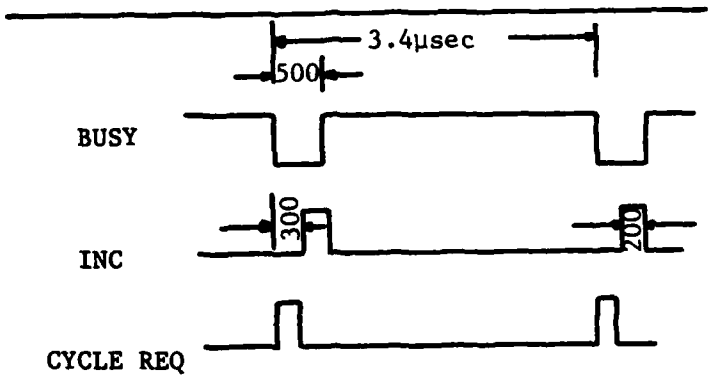


Figure F-9. Write mode timing diagram.

P1				P2			
13-37	KK	A001N	18	13-48	DD	END CYCLE	39
11-44	U	DATI07	16	49	BB	CYCLE REQB	37
43	R	DATI06	14	50	Z	FNCT 30	35
42	N	DATI05	12				33
41	L	DATI04	10	51	Y	FNCT20	31
40	J	DATI03	8	52	U	FNCT10	29
39	E	DATI02	6				27
38	C	DATI01	4	53	HH	0+5	25
12-39	A	DATI00	2	54	S	READY0	23
		CYCLE REQ A	39	55	P	WC INC ENB	21
13-38	NN	GO0	37	13-56	m	INIT0	19
39	X	DSTATA	35	10	K	0DAT08	15
40	DD	SINGLE CYCLE	33	11	f	0DAT09	13
41	BB	DSTATB	31	12	e	0DAT10	11
42	Z	DSTATC	29	13	H	0DAT11	9
43	U	C1 CONTROL	27	14	C	0DAT12	7
44	HH	BUSY 0	25	15	a	0DAT13	5
		SPARE	23	16	Y	0DAT14	3
45	S	BA INC ENB	21	13-17	W	0DAT15	1
46	P	ATTN	19	13-57	KK	NO LOCK	28
13-47	M	CO CONTROL IN	17	13-9	U	0DAT07	16
12-45	K	DATI08	15	8	R	0DAT06	14
46	f	DATI09	13	7	N	0DAT05	12
47	e	DATI10	11	6	L	0DAT04	10
48	H	DATI11	9	5	J	0DAT03	8
49	C	DATI12	7	4	E	0DAT02	6
50	a	DATI13	5	3	C	0DAT01	4
51	Y	DATI14	3	13-2	A	0DAT00	2
12-52	W	DATI15	1				

Figure F-10. Interface and control connector.

BOX GEN		IRIG A/N	SIGNAL
A	RED	A	HD2
B	YEL	B	HD1
C	RED	C	TD8
D	GRE	D	TD4
E	RED	E	TD2
F	BLU	F	TD1
J	RED	H	UD8
K	BLK	J	UD4
L	RED	K	UD2
M	WHT	L	UD1
N	RED	M	TH2
P	BRO	N	TH1
R	RED	P	UH8
S	ORA	R	UH4
U	GRE	S	UH2
V	BRO	T	UH1
KK	WHT	U	TM4
LL	BRO	V	TM2
W	GRE	W	TM1
X	YEL	X	UM8
Y	GRE	Y	UM4
Z	ORA	Z	UM2
a	GRE	AA	UM1
b	WHT	AB	TS4
c	GRE	AC	TS2
d	BLU	AD	TS1
H	YEL	AE	US8
T	BLU	AF	US4
e	GRE	AH	US2

BOX GEN		IRIG P/N	SIGNAL
h	BLK	AJ	US1
f	BLU	AK	ts8
J	SHT	AL	ts4
k	BLU	AM	ts2
L	ORA	AN	ts1
M	BLU	AP	hs8
N	BRO	AR	hs4
P	BLU	AS	hs2
R	BLK	AT	hs1
S	YEL	AU	ms8
T	BRO	AV	ms4
HH	WHT	AW	ms2
JJ	BLK	AX	ms1
U	YEL	BF	SAMPLE TIME
V	BLK	BJ	GND
X	YEL	BK	GND
Y	WHT	BL	GND
Z	ORA	BM	GND
AA	BRO	BN	GND
NN	BRO	BW	SYNC ERROR
BB		CA	HD8
CC		CB	HD4
DD		CC	TH8
EE		CD	TH4

Figure F-11. Time code generator cable (J2)

APPENDIX G

EXAMPLE RESULTS

Once the ground truth on the sensor location and the locations of all targets within the imagery has been established, a variety of statistics can be computed from the operator responses. Typically, the specific analysis and formats for graphical outputs would be provided by the user in a user supplied program. To illustrate the types of data analysis that can be performed, results from one phase of the FOG-M experiment have been included in this appendix. No attempt should be made to draw any conclusions from these specific graphs or computer pientents.

Figures G-1, G-2, and G-3 show the classical number of detections, recognitions, and identifications as a function of range to the target in the histogram format. This type of output is available from a high resolution type of dot matrix printer/plotter. Figure G-4 is a sample computer printout of the different types of data and analysis used in this particular experiment.

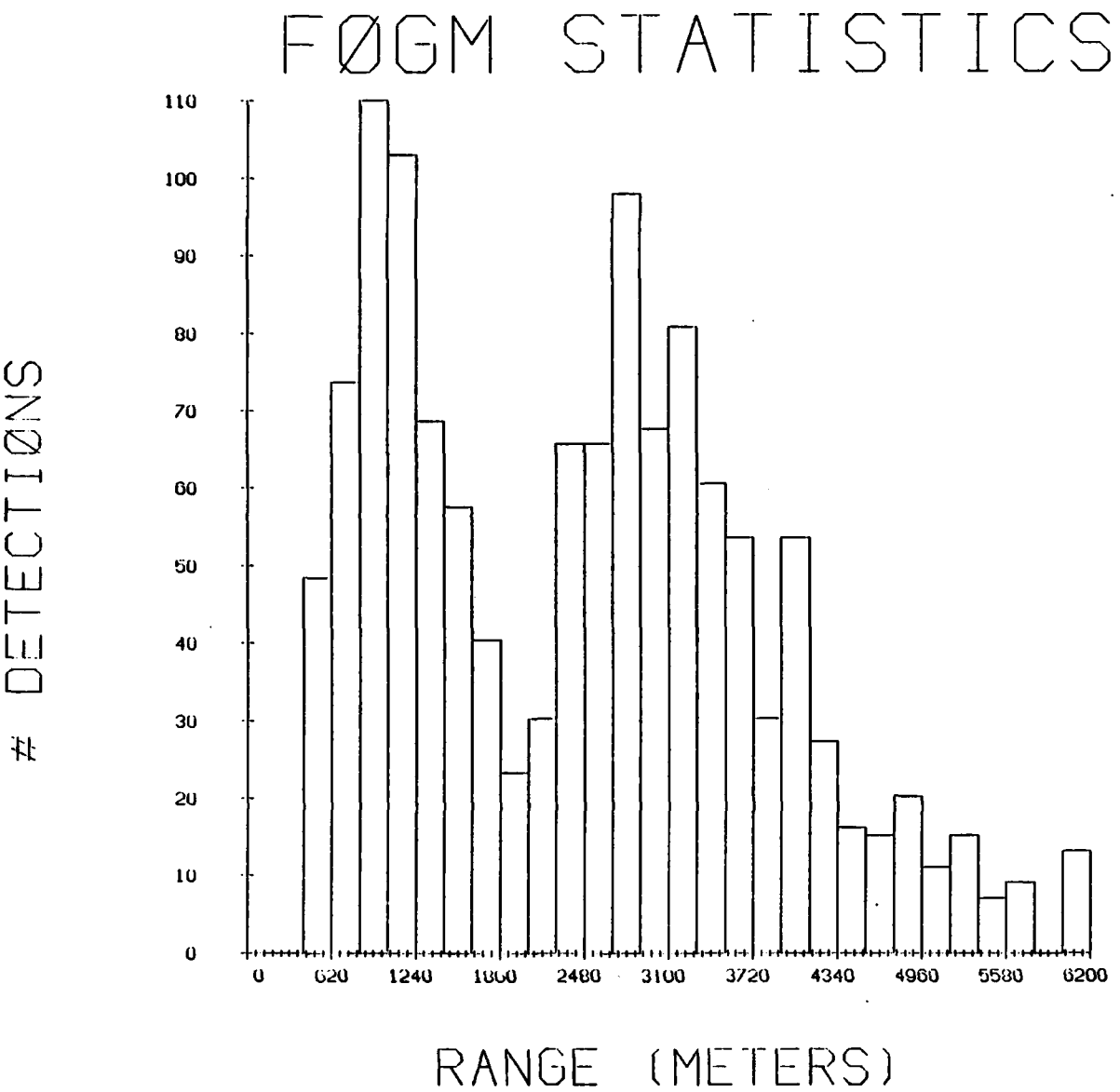

FØGM STATISTICS
DETECTIØNS
110
100
90
80
70
60
50
40
30
20
10
0
0
620
1240
1860
2480
3100
3720
4340
4960
5580
6200
RANGE (METERS)

FIG
G-1

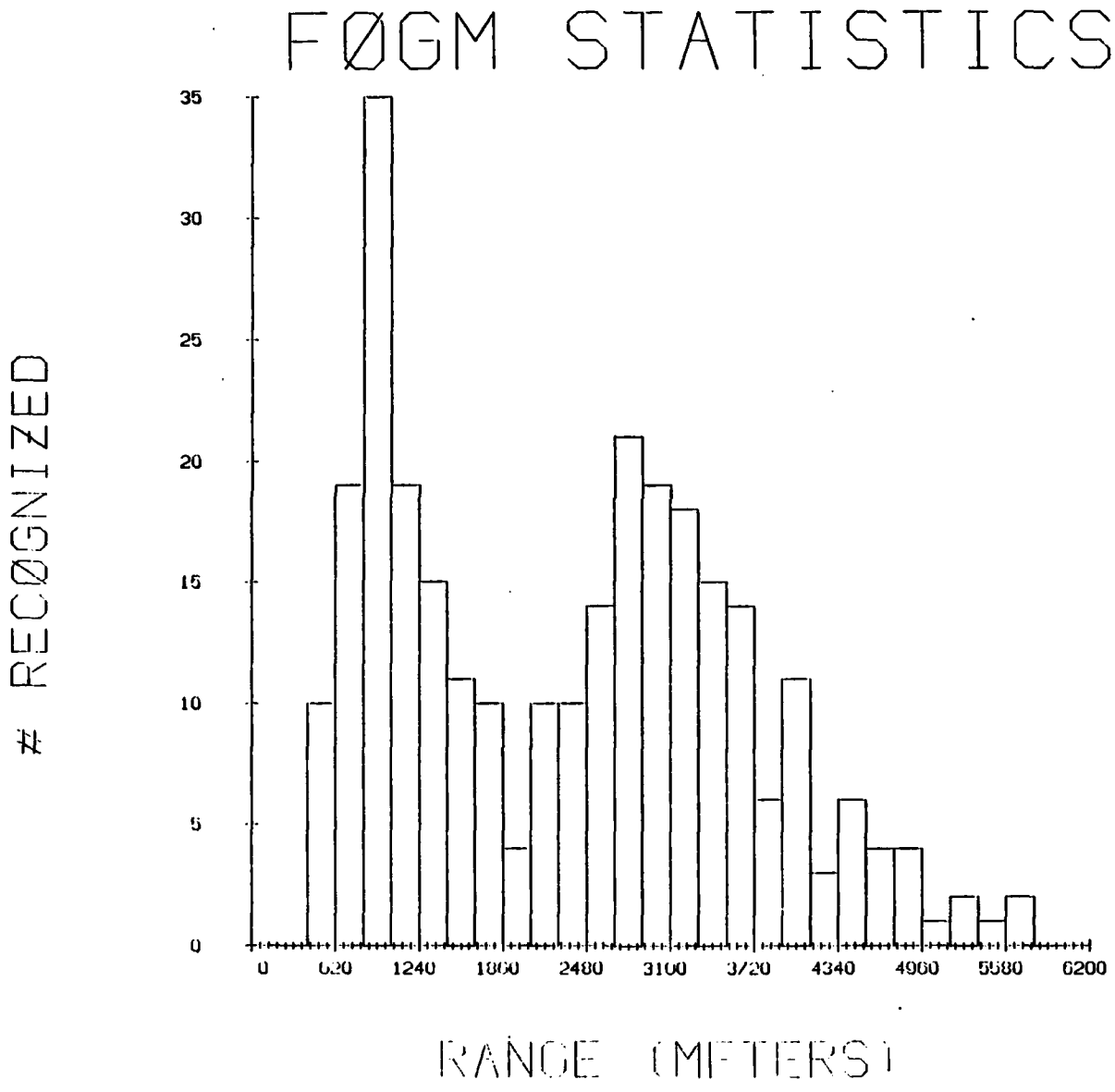

FØGM STATISTICS
RECØGNIZED
35
30
25
20
15
10
5
0
0
620
1240
1860
2480
3100
3720
4340
4960
5580
6200
RANGE (METERS)

FIG 6-2

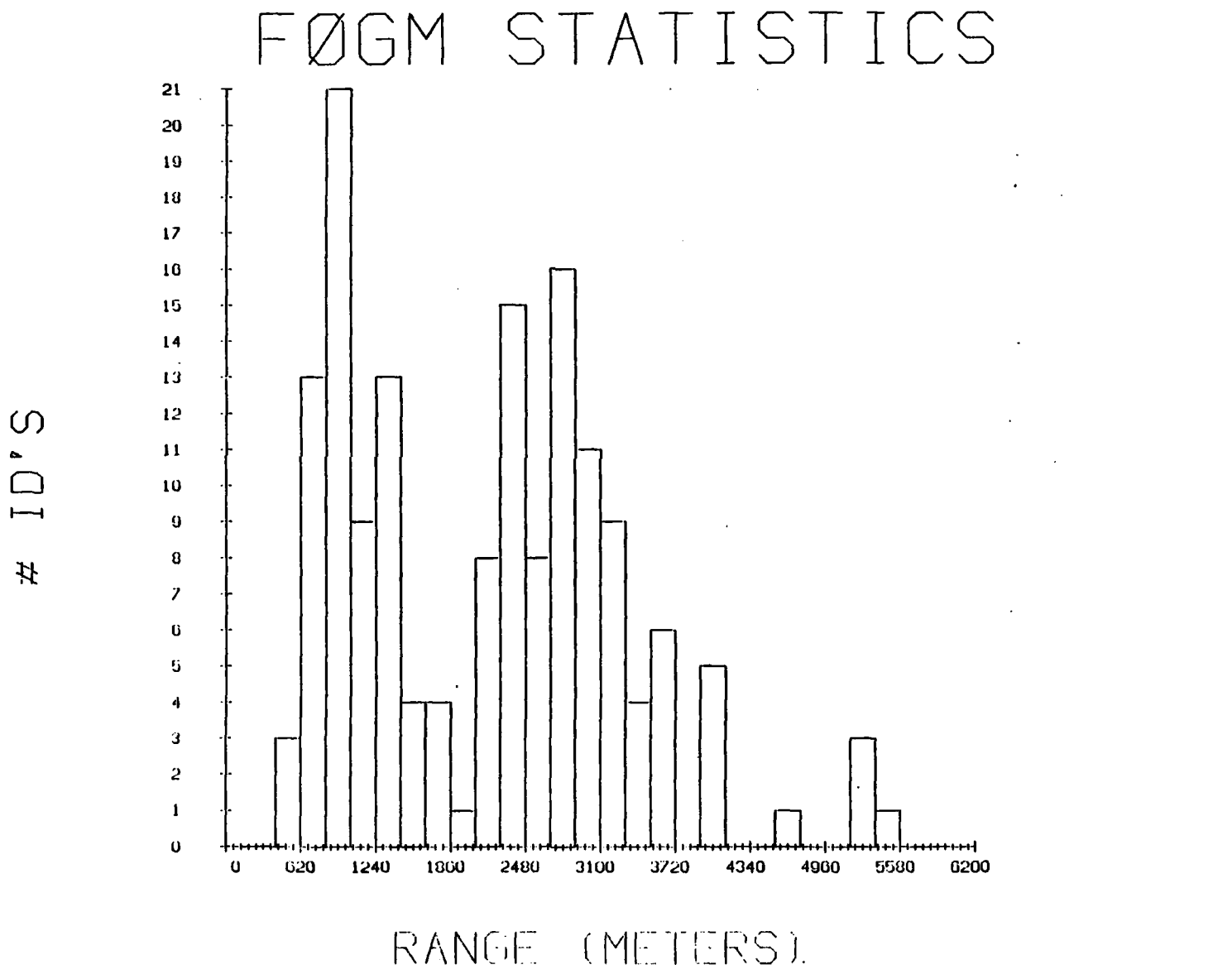

FØGM STATISTICS
ID'S
21
20
19
18
17
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
0
620
1240
1860
2480
3100
3720
4340
4960
5580
6200
RANGE (METERS)

FIG 6-7

```
$RUN AALT
 READING GROUND TRUTH DATA FILE GROUND.FFF.. STANDBY.
TOTAL NUMBER OF IRIGS=        11861
 READING UNFORMATTED MINI RANGER DATA FROM FILE MINI.DAT
       6835 MINI RANGER DATA READ SO LETS GET ON WITH IT.
NUMBER OF CORRECT TANK  RECOGNITIONS= 457     WRONG=   41
NUMBER OF CORRECT APC   RECOGNITIONS=  66     WRONG=   10
NUMBER OF CORRECT TRUCK RECOGNITIONS=  17     WRONG=   31
NUMBER OF CORRECT JEEP  RECOGNITIONS=  14     WRONG=    6

NUMBER OF CORRECT M60   IDENTIFICATIONS=   6 WRONG=   69
NUMBER OF CORRECT M48   IDENTIFICATIONS=  39 WRONG=   80
NUMBER OF CORRECT M551  IDENTIFICATIONS=  89 WRONG=   23
NUMBER OF CORRECT M113  IDENTIFICATIONS=  22 WRONG=    8
NUMBER OF CORRECT LANCE IDENTIFICATIONS=  10 WRONG=   13
NUMBER OF RESETS=      297
NUMBER OF OPERATORS=      26
NUMBER OF TOTAL OPERATOR RESPONSES=      4327
NUMBER OF WRONG TARGET RECOGNITIONS    ON ACTUAL TARGETS=         88
NUMBER OF WRONG TARGET IDENTIFICATIONS ON ACTUAL TARGETS=        193
NUMBER OF FALSE TARGET RECOGNITIONS    =         261
NUMBER OF FALSE TARGET IDENTIFICATIONS=           92
 DESIGNATIONS FOR FALSE TARGET SEQUENCE  4=   69
 DESIGNATIONS FOR    NO TARGET SEQUENCE 13=   30
 DESIGNATIONS FOR    NO TARGET SEQUENCE 17=   62
     SEQUENCE  1 DESIGNATION COUNT=         293
     SEQUENCE  2 DESIGNATION COUNT=         188
     SEQUENCE  3 DESIGNATION COUNT=         128
     SEQUENCE  4 DESIGNATION COUNT=           0
     SEQUENCE  5 DESIGNATION COUNT=         283
     SEQUENCE  6 DESIGNATION COUNT=         286
     SEQUENCE  7 DESIGNATION COUNT=         290
     SEQUENCE  8 DESIGNATION COUNT=         457
     SEQUENCE  9 DESIGNATION COUNT=         279
     SEQUENCE 10 DESIGNATION COUNT=         400
     SEQUENCE 11 DESIGNATION COUNT=         434
     SEQUENCE 12 DESIGNATION COUNT=         121
     SEQUENCE 13 DESIGNATION COUNT=           0
     SEQUENCE 14 DESIGNATION COUNT=         300
     SEQUENCE 15 DESIGNATION COUNT=         202
     SEQUENCE 16 DESIGNATION COUNT=           0
     SEQUENCE 17 DESIGNATION COUNT=           0
     SEQUENCE 18 DESIGNATION COUNT=         137
OUTSIDE GROUND TRUTH IRIG=        394
MEAN DETECT TIME=  2.914 SECONDS ON   777 DETECTIONS
WITH A STANDARD DEVIATION OF  2.526
MEAN RECOGNITION TIME=     1.233 SECONDS ON     539 TARGETS
WITH A STANDARD DEVIATION OF      1.727
MEAN IDENTIFICATION TIME=     1.808 SECONDS ON     151 TARGETS
WITH A STANDARD DEVIATION OF      1.921

IHCOUNT=          1253 IPLOT=           1
NUMBER OF DETECTIONS AT 325 FEET =         420
NUMBER OF DETECTIONS AT 500 FEET =         275
NUMBER OF DETECTIONS AT 800 FEET =         558
NUMBER OF DETECTIONS OUT OF THE ABOVE ALTS=          0
 ERROR ALLOWABLE IN MINIRANGER DATA =    1.500000

IHCOUNT=           284 IPLOT=           2
NUMBER OF RECOGNITIONS AT 325 FEET =         102
NUMBER OF RECOGNITIONS AT 500 FEET =          61
NUMBER OF RECOGNITIONS AT 800 FEET =         121

IHCOUNT=           155 IPLOT=           3
NUMBER OF IDENTIFICATIONS AT 325 FEET =         62
NUMBER OF IDENTIFICATIONS AT 500 FEET =         28
NUMBER OF IDENTIFICATIONS AT 800 FEET =         65
NUMBER OF IDENTIFICATIONS OUT OF THE ABOVE ALTS=
 ERROR ALLOWABLE IN MINIRANGER DATA =    1.500000
```

Figure G-4. Sample computer printout.

DISTRIBUTION

	NO. OF COPIES
Commander US Army Electronics Command Night Vision and E-O Laboratory ATTN: DELNV-AC, Mr. T. Jones Ft. Belvoir, VA 22060	2
Commander US Army Training and Doctrine Command Ft. Monroe, VA 23341	1
IIT Research Institute ATTN: GACIAC 10 West 35th Street Chicago, IL 60616	1
Defense Advanced Research Projects Agency Tactical Technology 1400 Wilson Blvd Arlington, VA 22209	1
Commander US Air Force Armanent Laboratory Eglin AFB, FL 32542	1
Commander US Army TRADOC Systems Analysis Activity ATTN: ATAA-TDS White Sands Missile Range, NM 88002	1
Director US Army Human Engineering Laboratory ATTN: DRXHE Aberdeen Proving Ground, MD 21005	1
Commander US Army Materiel Analysis Activity ATTN: DRXSY-MP Aberdeen Proving Ground, MD 21005	1

DISTRIBUTION

		COPIES
DRXHE-MI,	Mr. Nichols	2
AMSMI-R,	Dr. McCorkle	1
	Dr. Rhoades	1
	Mr. Cobb	1
AMSMI-RE,	Mr. Todd	1
-RES,		10
-REO,	Mr. Crosswhite	2
-REI		1
-RG,	Mr. Jacobs	1
-RGC		1
-RGN		1
-RD,	Mr. Holder	1
	Mr. Powell	1
-RPT		1
-RPR		15
-LP		1

END

FILMED

11-85

DTIC